DOCTOR OF PHILOSOPHY

Extensions in Non-Minimal State-Space and State-Dependent Parameter Model Based Control with Application to a DC-DC Boost Converter

Hitzemann, Ulrich

Award date:
2013

Awarding institution:
Coventry University

Extensions in Non-Minimal State-Space and State-Dependent Parameter Model Based Control with Application to a DC-DC Boost Converter

Ulrich Hitzemann

Dipl.-Ing.(FH) Mechatronik
MSc Systems and Control

A thesis submitted in partial fulfilment of the University's requirements for the degree of Doctor of Philosophy

September 2013

Control Theory and Applications Centre
Coventry University

Abstract

This Thesis is concerned with model-based control, where models of linear non-minimal state-space (NMSS) and nonlinear state-dependent parameter (SDP) form are considered. In particular, the focus is on model-based predictive control (MPC) in conjunction with the linear NMSS model and on proportional-integral-plus (PIP) pole-assignment control in conjunction with the SDP model.

The SDP-PIP pole-assignment controller is based on a nonlinear SDP model, however, the approach uses a linear pole-assignment controller design technique. This 'potential paradox' is addressed in this Thesis. A conceptual approach to realising the SDP-PIP pole-assignment control is proposed, where an additional conceptual time-shift operator is introduced. This allows the SDP-PIP, at each sampling time instance, to be considered as an equivalent linear controller, while operating, in fact, in a nonlinear overall context. Additionally, an attempt to realise SDP-PIP control, where the SDP model exhibits equivalent linear system numerator zeros, is proposed.

Regarding the NMSS MPC, emphasis is on square, i.e. equal number of inputs and outputs, multi-input multi-output (MIMO) modelled systems, which exhibit system output cross-coupling effects. Moreover, the NMSS MPC in incremental input form and making use of an integral-of-errors state variable, is considered. A strategy is proposed, that allows decoupling of the system outputs by diagonalising the closed-loop system model via an input transformation. A modification to the NMSS MPC in incremental input form is proposed such that the transformed system input - system output pairs can be considered individually, which allows the control and prediction horizons to be assigned to the individual pairs separately. This modification allows imposed constraints to be accommodated such that the cross-coupling effects do not re-emerge.

A practical example is presented, namely, a DC-DC boost converter operating in discontinuous conduction mode (DCM), for which a SDP model is developed. This model is based on measured input-output data rather than on physical relationships. The model incorporates the output current so that the requirements for the load, driven by the converter, is constrained to remain within a predefined output current range. The proposed SDP model is compared to an alternative nonlinear Hammerstein-bilinear structured (HBS) model. The HBS model is, in a similar manner to the SDP model, also based on measured input-output data. Moreover, the differences as well as the similarities of the SDP and HBS model are elaborated. Furthermore, SDP-PIP pole-assignment control, based on the developed SDP model, is applied to the converter and the performance is compared to baseline linear PIP control schemes.

Acknowledgements

First, I would like to thank my director of studies, Professor Keith Burnham, for providing me the opportunity to carry out my PhD studies and for his support during this time.

I am also grateful for the support of the people within the Control Theory and Applications Centre and the enjoyable time I could spend with them during the years I stayed in Coventry.

Contents

List of Figures

List of Tables

Abbreviations

ADC	Analog to Digital Converter
CCM	Continuous Conduction Mode
CLP	Closed-Loop Paradigm
DAC	Digital to Analog Converter
DCM	Discontinuous Conduction Mode
FIS	Fixed Interval Smoothing
GPC	Generalised Predictive Control
HBS	Hammerstein-Bilinear Structure
KKT	Karush-Kuhn-Tucker
LMFD	Left Matrix Fraction Description
LP	Linear Programming
LQ	Linear Quadratic
LQG	Linear Quadratic Gaussian
LS	Least Squares
MC	Monte-Carlo
MIMO	Multi Input - Multi Output
MPC	Model-based Predictive Control
NMSS	Non-Minimal State-Space

PI	Proportional-Integral
PID	Proportional-Integral-Derivative
PIP	Proportional-Integral-Plus
PWM	Pulse Width Modulated
QP	Quadratic Programming
RLS	Recursive Least Squares
SDP	State Dependent Parameter
SDRE	State Dependent Riccati Equation
SeDP	Semidefinite Programming
SISO	Single Input - Single Output
TVP	Time Varying Parameter

Chapter 1

Introduction

1.1 Motivation

In the real world, most systems, whether man-made or natural, are nonlinear. However, in some cases, the nonlinearities can be considered negligible, especially when some systems are intended to operate about a fixed operating point. In such cases, linearisation around this operating point is often sufficient in practice. As a consequence, linear control techniques are still in high demand, due mainly to their performance, yet practicability and ease of implementation.

In particular, the non-minimal state-space (NMSS) approach offers a further simplification in terms of control design, since all the system states are measurable, i.e. current and previous system output values as well as previous system input values, which removes the need for a state observer/estimator (Hesketh 1982; Young et al. 1987).

Inspired by the above mentioned useful and practical properties, model-based control strategies in the NMSS framework have been explored. Particularly, NMSS model-based predictive control (MPC), where the NMSS MPC in incremental input form (Wang and Young 2006) and making use of an integral-of-errors

state variable (Exadaktylos et al. 2006), in order to ensure offset free set-point tracking, is considered. This approach is attractive when multiple-input multiple-output (MIMO) systems are considered, which exhibit output cross-coupling effects.

Although linear control methodologies are able to provide appropriate solutions to a variety of control problems, there are limitations and often compromises are required to be made. Consequently, a step towards considering nonlinear control becomes a necessity. Nonlinear model-based control, that makes use of state-dependent parameter (SDP) models (Young et al. 2001), allows the use of linear control techniques in a nonlinear SDP framework. Particularly, the focus is on SDP pole-assignment control (Taylor et al. 2009), which can be viewed as an extension of the linear proportional-integral-plus (PIP) controller (Young et al. 1987; Wang and Young 1988), i.e. SDP-PIP. The implications of applying pole-assignment, i.e. a linear control technique, in a nonlinear domain has not been explored/reported in detail and this is one of the issues addressed in this Thesis, where further clarification is provided.

It may be argued that the best way of confirming the applicability and usefulness of a theoretical control method is to apply it to a practical, real-world system. In this regard, a SDP-PIP controller is applied to a purpose built laboratory-based DC-DC boost converter, which is operated in discontinuous conduction mode (DCM), and practical experiments are conducted. Additionally, since SDP-PIP is model-based, an SDP model of the converter has been developed. Although there are several approaches to modelling and control of a DC-DC boost converter, to the authors knowledge, the SDP framework has not been used on such an application before.

1.2 Thesis outline

The methodological background concepts, upon which this Thesis is based, as well as the methods/techniques, that are used in this Thesis, are introduced in Chapter 2. This comprises the concept of NMSS and SDP system representations as well as respective system identification methods. The concept of MPC and algorithms which allow the handling of imposed constraints, and finally, the concept of PIP pole-assignment control is also introduced.

In the context of this Thesis, reviews of the relevant previous developments regarding the NMSS and SDP methodology, the corresponding MPC and PIP control approaches as well as modelling and control strategies for DC-DC converters are given in the introductory parts of the respective Chapters.

Chapter 3 is concerned with a decoupling NMSS MPC approach for linear MIMO systems such that system output cross-coupling effects are suppressed. The NMSS MPC formulations in incremental input form (Wang and Young 2006), as well as using an integral-of-errors state variable form (Exadaktylos et al. 2006), are considered. The approach presented also extends to the handling of imposed constraints.

In Chapter 4, discrete-time PIP pole-assignment control applied to nonlinear SDP system models is considered (Taylor et al. 2009). Emphasis is placed on the linear pole-assignment technique in a nonlinear SDP context. Moreover, SDP-PIP control of SDP modelled systems with equivalent system zeros is also considered and the difficulties arising from system-model mismatch are highlighted.

Subsequently, in Chapter 5, a SDP model for a DC-DC boost converter operating in DCM is developed and compared with an alternative Hammerstein-bilinear structured (HBS) system modelling approach. Results are verified by

making use of laboratory based experiments with the DC-DC converter.

Based upon the developed SDP model in Chapter 5, implementation results of SDP-PIP control are presented in Chapter 6. The implementation results obtained by making use of the SDP-PIP are compared with baseline linear PIP control implementation results when tuned for operating about a particular fixed point.

Finally, conclusions are presented and directions for further work are suggested in Chapter 7.

1.3 Contributions

In this Section, the findings and contributions presented in this Thesis are summarized, in the order as it appears in the Thesis:

- *Decoupling MIMO NMSS MPC - Chapter 3.* An approach is presented for an analytic system output decoupling technique (Plummer and Vaughan 1997), applied to MPC based on square (equal number of inputs and outputs) MIMO models in a NMSS representation. This approach makes use of a closed-loop system diagonalisation method such that cross-coupling effects from the set-point command signals to the system outputs are suppressed. Moreover, handling of imposed constraints is considered and the effects on the MPC formulation in incremental input form (Wang and Young 2006) as well as using an integral-of-errors state variable (Exadaktylos et al. 2006) are explored. In particular, a further modification to the MPC in incremental input form is presented, which allows the cross-coupling elimination/suppression to be maintained when constraints are imposed. This work has been partially published in:

U. Hitzemann and K. J. Burnham. Decoupling model predictive control in a non-minimal state space representation. In *Proceedings of the 8th European Workshop on Advanced Control and Diagnosis, ACD 2010*, Ferrara, Italy, 2010

- *Developments in SDP-PIP pole-assignment control - Chapter 4.* Since a model-based control approach, i.e. SDP-PIP (Taylor et al. 2009), is considered, in this context, the SDP model itself is explored. Furthermore, a framework is presented, where an additional conceptual time-shift operator is introduced in order to take into account the evolution of the nonlinear SDP-PIP controlled system, while the standard and commonly used time-shift operator considers only the sampling time-step instantaneous linear model, upon which the discrete-time controller gains are determined. This provides further clarity of interpretation. Moreover, the SDP-PIP applied to SDP represented systems, which exhibit equivalent system zeros is also addressed.

- *SDP modelling of a DC-DC boost converter operating in DCM - Chapter 5.* The remainder of the Thesis is concerned with an example of a practical application; namely a purpose built laboratory based DC-DC boost converter. A SDP modelling approach for a DC-DC boost converter, which operates in DCM, is presented. This SDP model is compared to a HBS modelling approach. This work has been partially published in:

  T. Larkowski, U. Hitzemann and K. J. Burnham. Modelling and Identification of a DC-DC Boost Converter Operating in Discontinuous Conduction Mode. In *Proceedings of the IET Control and Automation Conference*, Birmingham, UK, 2013

  U. Hitzemann and K. J. Burnham. State Dependent Parameter Modelling of a DC-DC Boost Converter Operating in Discontinuous Conduction Mode. In *Proceedings of the 9th International Conference on Inform-*

atics in Control, Automation and Robotics, ICINCO 2012, pages 482–487, Rome, Italy, 2012

- *SDP-PIP control of a DC-DC boost converter operating in DCM - Chapter 6.* Based upon the developed SDP model, a SDP-PIP controller is applied and compared to two forms of linear PIP controllers, tuned for a fixed operating point, which are based on a linearised model and an instantaneous linear model. The operating point that these linear models are obtained for, is chosen such that it lies centrally within the operating range. Regulation of load steps, as well as output voltage set-point changes are considered. This work has been partially published in:

  U. Hitzemann and K. J. Burnham. State Dependent Parameter Modelling and Control of a DC-DC Boost Converter in Discontinuous Conduction Mode. In *Proceedings of the 9th European Workshop on Advanced Control and Diagnosis, ACD 2011*, paper 56, Budapest, Hungary, 2011

Chapter 2

Background Concepts

In this Chapter, the methodological background concepts, of which use is made in this Thesis, are introduced. Generally, the conceptual approach is presented in this Chapter only, since these concepts are taken from textbooks and other publications so that detailed information can be found in the respective references provided and further references therein.

The system models considered in this Thesis are in discrete-time domain and mainly in a discrete-time difference equation form, hence the focus builds on this structure. Based on this model structure and under the consideration of a linear system model, in Section 2.1, the concept of a non-minimal state-space (NMSS) system representation is depicted.

Extending this concept to a nonlinear framework, in Section 2.2, the state-dependent parameter (SDP) system representation is introduced.

Subsequently, the system identification methods used in order to identify the, essentially unknown, model parameters of the linear NMSS, as well as the nonlinear SDP model are presented in Section 2.3.

Furthermore, in this Thesis, two model-based control strategies are considered. Section 2.4 is concerned with a model-based predictive control (MPC)

approach, where the model on which the MPC is based, is in a NMSS form, hence linear. Moreover, the handling of imposed constraints is also considered.

The second model-based control approach considered, is the proportional-integral-plus (PIP) controller, which is introduced in Section 2.5. Here, the PIP controller based on the linear NMSS and based on the nonlinear SDP model, is considered.

2.1 Non-minimal state-space system representation

Consider the linear, discrete-time, single-input single-output (SISO) system model in difference equation form

$$\begin{aligned} y_k + a_1\ y_{k-1} + a_2\ y_{k-2} + \ldots + a_{n_a}\ y_{k-n_a} \\ = b_1\ u_{k-1} + b_2\ u_{k-2} + \ldots + b_{n_b}\ u_{k-n_b} \end{aligned} \tag{2.1}$$

where the subscript k denotes the sampling time instance and a_i, $b_i\ \in \mathbb{R}$ denote the model parameters, respectively. The system output and input are denoted by y and u, respectively. Alternatively, (2.1) can formulated in a linear, discrete-time transfer function of the form

$$y_k = \frac{B(z^{-1})}{A(z^{-1})}\ u_k \tag{2.2}$$

with

$$A(z^{-1}) = 1 + a_1\ z^{-1} + a_2\ z^{-2} + \ldots + a_{n_a}\ z^{-n_a} \tag{2.3a}$$

$$B(z^{-1}) = b_1\ z^{-1} + b_2\ z^{-2} + \ldots + b_{n_b}\ z^{-n_b} \tag{2.3b}$$

where z^{-1} denotes the backward time-shift operator, i.e. $z^{-1}\, y_k = y_{k-1}$. In general, a time delay can be introduced by setting the respective leading parameters in (2.3b) to zero.

The NMSS representation of the system can be directly deduced from the discrete-time transfer function or from the difference equation (Young et al. 1987), i.e. (2.2) and (2.1), respectively. This follows from the definition of the state vector of the NMSS model in which elements consists of the current and previous system output measurements and previous system inputs, see e.g. (Young et al. 1987; Wang and Young 2006; Wang 2009), i.e.

$$\boldsymbol{x}_k = [y_k \; y_{k-1} \; \cdots \; y_{k-n_a+1} \; u_{k-1} \; u_{k-2} \cdots \; u_{k-n_b+1}]^T \; \in \mathbb{R}^{(n_a+n_b-1)\times 1} \tag{2.4}$$

so that the NMSS system representation becomes

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{G}\; \boldsymbol{x}_{k-1} + \boldsymbol{B} u_{k-1} \\ y_k &= \boldsymbol{C}\; \boldsymbol{x}_k \end{aligned} \tag{2.5a}$$

where the $(n_a + n_b - 1) \times (n_a + n_b - 1)$ state transition matrix is

$$\boldsymbol{G} = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{n_a-1} & -a_{n_a} & b_2 & \cdots & b_{n_b-1} & b_{n_b} \\ 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \tag{2.5b}$$

and

$$\begin{aligned} \boldsymbol{B} &= [b_1 \ 0 \ 0 \ \cdots \ 0 \ 1 \ 0 \ \cdots \ 0]^T \ \in \mathbb{R}^{(n_a+n_b-1)\times 1} \\ \boldsymbol{C} &= [1 \ 0 \ \cdots \ 0] \ \in \mathbb{R}^{1\times(n_a+n_b-1)} \end{aligned} \tag{2.5c}$$

In the case of a multi-input multi-output (MIMO) system, the dimension of the difference equation representation (2.1) increases accordingly and, consequently, so do the matrices involved in the NMSS representation, see also Chapter 3.

2.2 State-dependent parameter system representation

Similar to the linear NMSS system model representation (2.5), the SDP model can be represented in a NMSS formulation as well. While the model parameters of the linear NMSS representation are constant quantities, the model parameters of the SDP representation are dependent on the states of the NMSS state vector (2.4), i.e. the current and previous system outputs as well as the previous system inputs. Moreover, the SDP model parameters are not restricted to be dependent on the states only, they also can be dependent on further variables (Young 2000, 2011).

In a similar manner as the linear NMSS system model representation of (2.5), the SDP model in a NMSS formulation can also be deduced from a discrete-time difference equation (Young 2000),

$$\begin{aligned} y_k + a_1(\boldsymbol{\chi}_k)\ y_{k-1} + a_2(\boldsymbol{\chi}_k)\ y_{k-2} + \ldots + a_{n_a}(\boldsymbol{\chi}_k)\ y_{k-n_a} \\ = b_1(\boldsymbol{\chi}_k)\ u_{k-1} + b_2(\boldsymbol{\chi}_k)\ u_{k-2} + \ldots + b_{n_b}(\boldsymbol{\chi}_k)\ u_{k-n_b} \end{aligned} \tag{2.6}$$

where the state dependency on the non-minimal states in the vector (2.4) is

denoted by

$$\boldsymbol{\chi}_k = [\boldsymbol{x}_k^T \ \boldsymbol{\vartheta}_k^T]^T \tag{2.7}$$

and $\boldsymbol{\vartheta}_k$ denotes a vector comprised of some variables the SDP model parameters may also depend on.

Consequently, for the sake of brevity and simplicity, the parameters can be viewed as time varying, so that (2.6) becomes

$$\begin{aligned} y_k + a_{1,k}\ y_{k-1} + a_{2,k}\ y_{k-2} + \ldots + a_{n_a,k}\ y_{k-n_a} \\ = b_{1,k}\ u_{k-1} + b_{2,k}\ u_{k-2} + \ldots + b_{n_b,k}\ u_{k-n_b} \end{aligned} \tag{2.8}$$

where the subscript k indicates the time varying nature of the model parameters and subsequently, a non-minimal state-space formulation can be obtained, such as (2.5), in which the parameters are state dependent.

2.3 System identification

In the previous sections of this chapter, the system representations have been introduced. In this Section attention is given to system identification methods in order to obtain the associated model parameters. The system identification methods considered here, are those used in this Thesis or those which an underlying concept is adopted, in particular, in Chapter 5, where a SDP modelling approach of a DC-DC boost converter is proposed.

2.3.1 Linear system identification methods

Consider the linear system model in discrete-time, difference equation form (2.1). A straightforward method in order to obtain the parameters is the least-squares (LS) algorithm (Hsia 1977) and its recursive version, i.e. recursive least-squares

(RLS), which provides additional information on statistical properties (Hsia 1977; Söderström and Stoica 1989; Wellstead and Zarrop 1991; Ljung 1999).

Since the LS algorithm identifies the unknown model parameters by representing (2.1) in regression vector form, i.e.

$$y_k = \boldsymbol{\varphi}^T \, \boldsymbol{\theta\psi} \tag{2.9}$$

where $\boldsymbol{\varphi}^T = [-y_{k-1} \; -y_{k-2} \; \cdots \; -y_{k-n_a} \; u_{k-1} \; \cdots \; u_{k-n_b}]$ denotes the regression vector and $\boldsymbol{\theta\psi} = [a_1 \; a_2 \; \cdots \; a_{n_a} \; b_1 \; \cdots \; b_{n_b}]^T$ denotes the parameter vector, also parameters of nonlinear functions can be identified if the function is linear w.r.t. the parameters and can be expressed in the form (2.9), e.g. $n-th$ order polynomials, which is also used in Chapter 5.

2.3.2 SDP system identification methods

Identifying the parameters of a SDP system model is not as straightforward as identifying the parameters of a linear model, although the SDP and linear model considered here are both of identical structure, i.e. in a discrete-time difference equation form (2.1) and (2.8), respectively. The main difficulty, however, lies in the fact that the SDP model parameters are themselves unknown functions of the non-minimal states which are required to be identified.

In order to identify these functions, several approaches have been made, such as based on artificial neural networks (Akesson and Toivonen 2006). An efficient method, which also provides an underlying concept that is used in Chapter 5, is the recursive fixed interval smoothing (FIS) method (Young et al. 2001). The SDP model is initially viewed as a linear, time-varying parameter (TVP) model and identified by making use of recursive linear system identification methods. Subsequently, these parameters are ordered in a non-temporal manner, e.g. they

are ordered in ascending magnitude of the previous system output y_{k-1} (or any other variable in the vector $\boldsymbol{\chi}$) at which time instance the respective TVP was identified, so that the parameter can be obtained as a function of this variable, which yields consequently the SDP. Further details on the method of using FIS and extensions to this method can be found in (Young 2011, Chap. 11) and is readily implemented in a Matlab® toolbox, named CAPTAIN[1] (Young and Taylor 2012).

2.4 Constrained model-based predictive control

Model-based predictive control (MPC) is a wide and, over the past decades, active research area. Comprehensive surveys of the developments in this field are presented in, e.g. (Morari and Lee 1999; Bemporad and Morari 1999; Mayne et al. 2000). The success of the MPC approach, also beyond academia, is indicated by the fact that it has found its way to various industrial applications and that MPC controllers are commercially available nowadays (Qin and Badgwell 2003).

Although extensive research by various researchers has been undertaken, the development of the generalised predictive controller (GPC) (Clarke et al. 1987a,b) can be considered a 'milestone' in transfer function based MPC. Since this pioneering work, often system models in a state-space representation are used (Kwon and Han 2005), which allow a relatively straightforward incorporation of constraints into the MPC formulation, see e.g. (Maciejowski 2001). Hence MPC based on linear models turns into a nonlinear optimization problem once constraints are imposed.

The MPC considered here, is, in general, formulated as a cost function of

[1] see `www.lancaster.ac.uk/staff/taylorcj/tdc/download.php` or `http://captaintoolbox.co.uk` [both accessed 09/2013]

convex, quadratic form (Maciejowski 2001; Rossiter 2004; Camacho and Bordons 2007), e.g.

$$J = \sum_{i=1}^{N_p} (\boldsymbol{y}_{k+i|k} - \boldsymbol{r}_{k+i|k})^T \boldsymbol{Q}_i (\boldsymbol{y}_{k+i|k} - \boldsymbol{r}_{k+i|k}) + \sum_{j=0}^{N_c-1} \Delta \boldsymbol{u}_{k+j|k}^T \boldsymbol{\Lambda}_j \Delta \boldsymbol{u}_{k+j|k} \quad (2.10)$$

which is required to be minimised and where $\boldsymbol{y}_k = [y_{1,k} \; y_{2,k} \; \ldots \; y_{n_y,k}]^T$, $\boldsymbol{r}_k = [r_{1,k} \; r_{2,k} \; \ldots \; r_{n_y,k}]^T$ denote vectors of n_y system outputs and respective reference signals, $\boldsymbol{u}_k = [u_{1,k} \; u_{2,k} \; \ldots u_{n_u,k}]^T$ denotes a vector of n_u system inputs, $\boldsymbol{Q}_i \in \mathbb{R}^{n_y \times n_y}$ and $\boldsymbol{\Lambda}_j \in \mathbb{R}^{n_u \times n_u}$ denote positive definite and positive semidefinite weighting matrices, respectively, and $\Delta = 1 - z^{-1}$ denotes the difference operator. The subscript $k+i|k$ denotes the i-th prediction based at the current sampling time instance k. Consequently, the cost function contains N_p system output predictions, hence N_p denotes the prediction horizon, and $N_c - 1$ system input predictions, hence N_c denotes the control horizon.

Now, consider the issue of constraint handling. Minimising the cost function (2.10) w.r.t. the system input prediction sequence $\{\Delta \boldsymbol{u}_{k+i|k}\}$ $i = 0, 1, \ldots, N_c - 1$, can be regarded as the unconstrained case, consequently, the constrained case can be formulated as an optimisation problem

$$\begin{aligned} &\underset{\Delta \boldsymbol{u}_{k|k}, \Delta \boldsymbol{u}_{k+1|k}, \ldots, \Delta \boldsymbol{u}_{k+N_c-1|k}}{min.} && J \\ &\quad subject\ to: && \boldsymbol{y}_{k+i} = f(\boldsymbol{y}_{k+i-j}, \boldsymbol{u}_{k+i-l}) \\ &&& \boldsymbol{y}_{k+i} \in \mathcal{Y} \quad i = 1, 2, \ldots, N_p \\ &&& \boldsymbol{u}_{k+i} \in \mathcal{U} \quad i = 0, 1, \ldots, N_c - 1 \end{aligned} \quad (2.11)$$

where $\mathcal{Y}$ and $\mathcal{U}$ denote constraint sets on the system output and input, respectively.

In the MPC approach, at every sampling time instance, a set of optimal,

current and predicted system inputs are calculated by obtaining the solution of the optimisation problem (2.11), i.e. $\{\boldsymbol{u}_{k|k},\ \boldsymbol{u}_{k+1|k},\ \ldots\ ,\ \boldsymbol{u}_{k+N_c-1|k}\}$, so that $N_c - 1$ future system inputs are obtained. The current system input $\boldsymbol{u}_k$, i.e. the first element of this sequence, however, is actually applied to the system and the remaining, predicted inputs are discarded. This procedure is repeated at each sampling instance. Therefore, the MPC method is also termed a receding horizon control (RHC) (Kwon and Han 2005). Additionally, the system input beyond the control horizon is assumed to be constant, i.e. $\boldsymbol{u}_{k+N_c+i|k} = \boldsymbol{u}_{k+N_c-1|k}\ \forall i = 0, 1, 2, \ldots$.

However, in (Bemporad et al. 2000, 2002) a multiparametric quadratic programming approach is proposed so that the optimisation problem (2.11) is not required to be solved online. This, however, is beyond the scope of this Thesis.

Furthermore, note that $N_p > N_c$. This becomes obvious when considering a linear system model (2.1) and the last element in the input prediction sequence, i.e. $u_{k+N_c-1|k}$, as well as the corresponding output prediction $y_{k+N_c|k}$. In the case of $N_c > N_p$, the input sequence is optimised beyond the prediction horizon and these additional predicted inputs do not affect the system output predictions that appear in the cost function (2.10), hence the choice of $N_p > N_c$.

Next, obtaining the solution of the optimisation problem (2.11) is of interest. As mentioned above, the optimisation problem (2.11) is convex and of a quadratic form, so that (2.11) can be cast as the following general quadratic optimisation problem

$$\begin{aligned} \underset{\boldsymbol{\theta}\in\mathbb{R}^{n\times 1}}{min.} \quad & f(\boldsymbol{\theta}) = \tfrac{1}{2}\boldsymbol{\theta}^T\boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{c}^T\boldsymbol{\theta}\boldsymbol{\psi} \\ subject\ to: \quad & \boldsymbol{A}_E\boldsymbol{\theta} = \boldsymbol{b}_E \\ & \boldsymbol{A}_I\boldsymbol{\theta} \leq \boldsymbol{b}_I \end{aligned} \tag{2.12}$$

where $\boldsymbol{A}_E \in \mathbb{R}^{n \times m_E}$, $\boldsymbol{A}_I \in \mathbb{R}^{n \times m_I}$ denote matrices and $\boldsymbol{b}_E \in \mathbb{R}^{m_E \times 1}$, $\boldsymbol{b}_I \in \mathbb{R}^{m_I \times 1}$ denote vectors that account for the equality and inequality constraints, respectively, $\boldsymbol{H} \in \mathbb{R}^{n \times n}$ denotes a positive definite matrix and $\boldsymbol{c} \in \mathbb{R}^{n \times 1}$ a constant vector. Due to the linear constraints and the positive definiteness of $\boldsymbol{H}$, (2.12) is a strictly convex, quadratic programming (QP) problem, for which a solution is global and unique (Goodwin et al. 2005, Chap. 2). Moreover, the solution of the problem (2.12), denoted $\boldsymbol{\theta}^*$, must satisfy the constraints as well as the (first-order necessary) Karush-Kuhn-Tucker (KKT) conditions (Goodwin et al. 2005)

$$\begin{aligned} \boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{c} + \boldsymbol{A}_E^T \boldsymbol{\lambda}_E + \boldsymbol{A}_I^T \boldsymbol{\lambda}_I &= \boldsymbol{0} \\ \boldsymbol{\lambda}_I &\geq \boldsymbol{0} \\ \boldsymbol{\lambda}_I^T (\boldsymbol{A}_I \boldsymbol{\theta} - \boldsymbol{b}_I) &= 0 \end{aligned} \tag{2.13}$$

where $\boldsymbol{\lambda}_E \in \mathbb{R}^{m_E \times 1}$ and $\boldsymbol{\lambda}_I \in \mathbb{R}^{m_I \times 1}$ denote vectors of Lagrange multipliers.

The area of optimisation is a wide field, ranging from linear programming (LP) methods (Hillier and Lieberman 2001) via QP methods (Fletcher 2000; Goodwin et al. 2005; Boyd and Vandenberghe 2004; Nocedal and Wright 2006) to semidefinite programming (SeDP) (Boyd et al. 1994) and numerous other methods. In particular, the SeDP method, developed by (Nemirovskii and Gahinet 1994; Gahinet and Nemirovski 1997), allows the efficient solution of linear matrix inequalities, which are used in the development of robust, constrained MPC methods (Kothare et al. 1996; Kouvaritakis et al. 2000, 2002). Also, this method is readily implemented in the Matlab® Robust Control Toolbox. However, this is beyond the scope of this Thesis and therefore, the focus here is on algorithms in order to solve QP problems. The most common algorithms in order to solve QP problems are the active set method and interior point methods (Maciejowski 2001). Matlab® provides the function `quadprog` in the Optimisation Toolbox,

where these algorithms are implemented and, in this Thesis, use is made of this function, in conjunction with the active set method, in order to obtain solutions of QP problems. In the subsequent sections, the active set method and interior point method implemented in the `quadprog` function are briefly described.

2.4.1 Active set method

In the active set method, as the name implies, the individual constraints in (2.12), i.e. $\boldsymbol{a}_i\boldsymbol{\theta} = b_i \ \forall i \in \mathcal{E}$ and $\boldsymbol{a}_i\boldsymbol{\theta} \leq b_i \ \forall i \in \mathcal{I}$, are either considered active or inactive, where $\mathcal{E} = \{1, 2, \ldots, m_E\}$, $\mathcal{I} = \{1, 2, \ldots, m_I\}$ denote sets consisting of indices corresponding to equality and inequality constraints, respectively, so that, consequently, $\boldsymbol{a}_i \ \forall i \in \mathcal{E}$ denotes the rows of $\boldsymbol{A}_E$ and $\boldsymbol{a}_i \ \forall i \in \mathcal{I}$ denotes the rows of $\boldsymbol{A}_I$, respectively, whereby $b_i \ \forall i \in \mathcal{E} \cup \mathcal{I}$ denote the respective elements of $\boldsymbol{b}_E$ and $\boldsymbol{b}_I$.

Suppose that the optimal, unconstrained solution of the QP problem $\tilde{\boldsymbol{\theta}}^*$ violates the imposed constraints, i.e. $\tilde{\boldsymbol{\theta}}^*$ lies outside the polytope formed by the constraints (Boyd and Vandenberghe 2004, Chap. 2) and is consequently not feasible. The solution of the QP problem with imposed constraints, however, lies as close as possible to the unconstrained solution, hence on the boundary of this polytope. This boundary consists of the equality constraints, but, additionally may also be formed by an inequality constraint, e.g. $\boldsymbol{a}_1\boldsymbol{\theta} \leq b_1$, which can be viewed, as a consequence, as an equality constraint $\boldsymbol{a}_i\boldsymbol{\theta} = b_i, \ i = 1 \in \mathcal{I}$, hence as an active constraint. Therefore, the active set, denoted $\mathcal{A}$, contains the indices of all equality constraints and of those inequality constraints, which can be regarded as equality constraints, i.e. $\mathcal{A} = \{i \in \mathcal{E} \cap \mathcal{I} \mid \boldsymbol{a}_i\boldsymbol{\theta} = b_i\}$.

Essentially, the active set method identifies the active constraints in an iterative manner, discards the inactive ones and transforms the QP problem with

inequality constraints into a QP problem with equality constraints only, i.e.

$$\begin{aligned} \underset{\boldsymbol{\theta}\in\mathbb{R}^{n\times 1}}{min.} \quad & f(\boldsymbol{\theta}) \\ subject\ to: \quad & \boldsymbol{a}_i\boldsymbol{\theta} = b_i \quad \forall i \in \mathcal{A} \end{aligned} \tag{2.14}$$

which can be easily solved by making use of the KKT condition (2.13), see e.g. (Nocedal and Wright 2006, Chap. 16.1).

In order to identify the active constraints, an initial feasible point is required, which is a problem in its own right and can be difficult to obtain, especially in large scale QP problems. However, since the aim is to minimise the cost function (2.12), a decreasing direction $\boldsymbol{d}_k \in \mathbb{R}^{n\times 1}$ is calculated at each iteration such that $f(\boldsymbol{\theta}_{k+1}) \leq f(\boldsymbol{\theta}_k)$ with $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \alpha_k \boldsymbol{d}_k$ where $\alpha \in (0, 1]$ denotes the step-length. If $\alpha_k = 1$ and $\boldsymbol{d}_k \neq \mathbf{0}$ does not yield a feasible solution, then a line search along $\boldsymbol{d}_k$ is performed, i.e. reducing α appropriately. Moreover, this also means that there exists a constraint in $\mathcal{I}$ along the direction $\boldsymbol{d}_k$, which is not considered in the active set. Consequently, the corresponding constraint index is obtained by finding the constraint index $i \notin \mathcal{A}$, that yields the smallest step length α_k, such that adding this constraint index to the active set, a feasible solution is obtained. A constraint becomes inactive (and removed from the active set), if it is a feasible point and the corresponding Lagrange multiplier in the KKT condition is negative. In the case of more than one negative Lagrange multiplier, the constraint index corresponding to the most negative one is removed from the active set. The algorithm terminates if $\boldsymbol{d}_k = \mathbf{0}$, the Lagrange multipliers are positive or zero, the solution is feasible and the KKT condition is satisfied. Detailed explanations on the active set method can be found in, e.g. (Fletcher 2000; Nocedal and Wright 2006).

2.4.2 Interior point method

Efficient interior point optimisation methods emerged initially for solving LP problems (Karmarkar 1984) and were later adopted for solving convex QP problems (Nesterov and Nemirovskii 1994; Renegar 2001). Furthermore, interior point methods are also used in order to solve optimisation problems where linear matrix inequalities are involved (Boyd et al. 1994), however, the focus here is on convex, QP problems.

Interior point methods start at an initial feasible point and iteratively converge to the solution of the QP problem. Other than the active set method, interior point methods search in the interior of the feasible region and not only on the boundary for the solution. However, it is not always straightforward to obtain an initial feasible point. This led to the development of infeasible interior point methods, which do not require an initial feasible starting point (Wright 1997). A drawback of these algorithms is that they cannot be terminated before convergence to the solution, since an intermediate point may be infeasible.

There is a wide variety of interior point algorithms, however, here an interior point method using barrier functions is considered. The barrier function describes the boundary of the feasible region, precisely, the boundary of the inequality constraints. Moreover, barrier functions are smooth, monotonically decreasing functions and are often of a logarithmic form, e.g.

$$\Phi(\boldsymbol{\theta}, \mu) = -\frac{1}{\mu} \sum_{i=1}^{m_I} \log(b_i - \boldsymbol{a}_i \boldsymbol{\theta}) \quad i \in \mathcal{I} \tag{2.15}$$

so that the QP problem (2.12) can be formulated to be

$$\begin{aligned} \underset{\boldsymbol{\theta}}{min.} \quad & f(\boldsymbol{\theta}) + \Phi(\boldsymbol{\theta}, \mu) \\ subject\ to: \quad & \boldsymbol{A}_E = \boldsymbol{b}_E \end{aligned} \tag{2.16}$$

which is a QP problem with equality constraints only, since the inequality constraints are now contained in the cost function, which is required to be minimised. Moreover, in the case of inequality constraints only, the problem is transformed into an, effectively, unconstrained QP problem. The parameter $\mu > 0 \in \mathbb{R}$ is increased in each iteration, i.e. $\mu \to \infty$, in order to avoid numerical issues, such as ill-conditioned matrices, when solving the equality constrained QP problem (2.16) at each iteration.

It is to be noted that the logarithmic barrier function method requires an initial feasible point, however, due to the barrier function, the algorithm stays within the feasible region and can be terminated before converging to the solution. Detailed information on this method can be found in, e.g. (Boyd and Vandenberghe 2004, Chap. 11).

2.5 Proportional-integral-plus control

The Proportional-Integral-Plus (PIP) control was initially developed in 1987 by (Young et al. 1987; Wang and Young 1988) as a pole-placement control approach. Therefore, it is a model-based controller which is based on a linear, NMSS model. The design of a pole-placement controller based on a NMSS model can also be found in earlier work by (Hesketh 1982). Other than making use of minimal state space models, the use of NMSS models makes the state observer obsolete since the states are measurable, hence the NMSS states are all observable.

Furthermore, the PIP controller can be viewed as an extension to the widely used proportional-integral-derivative (PID), in particular, the PI controller, see e.g. (Young et al. 1987).

Moreover, beside pole-placement design, also linear quadratic (LQ) and linear quadratic gaussian (LQG) optimal control designs have been developed,

see e.g. (Taylor et al. 1996a, 2000). Additionally, the PIP can be implemented based on the feedback and forward path structure, details on the differences can be found in (Taylor et al. 1996b).

Since Chapter 4 is concerned with the PIP feedback controller structure, this configuration is shown in Figure 2.1. The controller parameter polynomials

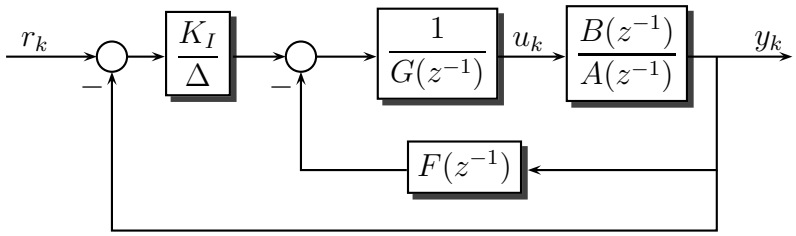


Figure 2.1: Block diagram of the linear PIP feedback structure

are denoted by $F(z^{-1})$ and $G(z^{-1})$, respectively, which are defined to be

$$\begin{aligned} F(z^{-1}) &= f_0 + f_1 z^{-1} + \ldots + f_{n_f} z^{-n_f} \quad n_f = n_a - 1 \\ G(z^{-1}) &= 1 + g_1 z^{-1} + \ldots + g_{n_g} z^{-n_g} \quad n_g = n_b - 1 \end{aligned} \tag{2.17}$$

and K_I denotes the integral gain, while $\Delta = 1 - z^{-1}$ denotes the discrete-time difference operator, i.e. $\Delta y_k = y_k - y_{k-1}$. From Figure 2.1, the control law is obtained to be

$$u_k = -F(z^{-1})y_k - \tilde{G}(z^{-1})u_k + \frac{K_I}{\Delta}(r_k - y_k) \tag{2.18}$$

with $\tilde{G}(z^{-1}) = G(z^{-1}) - 1$. Alternatively, in state-variable feedback form

$$u_k = -\boldsymbol{c}\ \hat{\boldsymbol{x}}_k^T \tag{2.19}$$

where $\boldsymbol{c} = [f_0 \cdots f_{n_f}\ g_1 \cdots g_{n_g}\ -K_I]$ and $\hat{\boldsymbol{x}}_k = [\boldsymbol{x}_k^T\ \zeta_k]^T$ denotes the augmented NMSS vector by the integral-of-errors state $\zeta_k = \frac{r_k - y_k}{\Delta}$. Note that in the case of

$n_a = n_b = 1$, the integral gain K_I and the gain f_0 are present only, which can be seen as a proportional gain so that, effectively, a PI controller is obtained. Also, from Figure 2.1, the closed-loop transfer function is given by

$$\frac{y_k}{r_k} = \frac{B(z^{-1})K_I}{\Delta\left(A(z^{-1})G(z^{-1}) + B(z^{-1})F(z^{-1})\right) + K_I B(z^{-1})} \tag{2.20}$$

The controller parameters are obtained by assigning the closed-loop poles, i.e.

$$D(z^{-1}) = \Delta\left(A(z^{-1})G(z^{-1}) + B(z^{-1})F(z^{-1})\right) + K_I B(z^{-1}) \tag{2.21}$$

where $D(z^{-1})$ denotes a predefined polynomial, which represents the desired characteristic equation of the closed-loop system, i.e. the denominator of (2.20), and by comparing coefficients of like powers of z yields the controller parameters. Alternatively, the state feedback gains are obtained by making use of optimal control techniques, such as LQ and LQG design, however, this is not considered in this Thesis. Further detailed information on the linear PIP can be found in the references given above in this Section and the references therein.

An attempt of using linear PIP control for nonlinear systems can be found in (McCabe et al. 2000) where feedback linearisation methods are used. In the following developments, in order to deal with nonlinear systems, the PIP is used in conjunction with SDP models to form the SDP-PIP, where at each sampling instance the SDP model is considered 'frozen'/instantaneous linear, so that linear control techniques can be applied, see e.g. (Kontoroupis et al. 2003; Taylor et al. 2009). Hence, the nonlinear SDP model is considered to be linear at each sampling instance, while, overall it is considered to be nonlinear. Consequently, these different 'levels', i.e. linear and nonlinear, of consideration are reflected in the formulation of the SDP-PIP controller in Chapter 4.

2.6 Concluding remarks

In this Chapter, the conceptual approaches and methodologies, for which this Thesis makes use, have been reviewed.

The linear, hence simplest, system representation in a NMSS form has been first considered, followed by its nonlinear extension, namely, the SDP system in NMSS representation.

Subsequently, system identification methods in order to identify the parameters of the previously considered system models, have been introduced. Both cases have been considered, the linear NMSS and nonlinear SDP system representations. In particular, the identification of parameters of SDP models using the method of FIS is emphasised since this method is adapted for the developments in Chapter 5.

Furthermore, the concept of linear predictive control, which is used in the developments of Chapter 3, has been introduced along with the handling of imposed constraints, i.e. convex QP optimisation algorithms, namely, the active set method and the interior point method based on logarithmic barrier functions.

Additionally, attention has been drawn to the PIP controller concept upon which the developments in Chapters 4 and 6 are based.

Chapter 3

Multivariable Decoupling NMSS MPC Control

This Chapter is concerned with an input-output decoupling control strategy for multivariable systems where the number of system outputs n_y is equal to the number of system inputs n_u, i.e. $n_y = n_u = n$.

Furthermore, an approach for system output decoupling is proposed making use of an input transformation term which diagonalises the closed-loop system, hence compensates for the cross-coupling effects. This is adapted from (Plummer and Vaughan 1997) and (Kubalcik and Bobal 2006) where pole assignment control is used. Here, however, model based predictive control is used whereby the model is of the non-minimal state-space form. Moreover, the NMSS-MPC controller in incremental input form (Wang and Young 2006) as well as that of the integral-of-errors state variable form (Exadaktylos et al. 2006) is used, and, in this context, their relative merits are evaluated. In particular, when imposing constraints, it is desired that the output decoupling is not impaired. In order to achieve this, a modification of the incremental input representation is proposed so that it is straightforward to obtain decoupled control of the system outputs,

despite the imposed constraints.

Existing output decoupling control methods are often either based on optimisation procedures, see e.g. (Lees et al. 1995; Sourlas 2001; Gunnarsson et al. 2003; Exadaktylos and Taylor 2010), aiming for minimising the cross-coupling effects, or by introducing compensation terms that ideally eliminate the cross-coupling effects completely by diagonalisation of the resulting closed-loop system.

A decoupling NMSS MPC strategy based on optimisation procedures is proposed in (Exadaktylos and Taylor 2010), which is based on a multi-objective optimisation approach, similar to (Bemporad and de la Pena 2009), using the goal attainment optimisation method, which is also used in (Lees et al. 1995) in a decoupling pole-assignment context. Most recent decoupling NMSS PIP developments in the continuous-time domain can be found in (Taylor et al. 2012). However, the discrete-time domain is considered in this Chapter only.

The method of closed-loop system diagonalisation by making use of a compensation term, in conjunction with pole-assignment control, can be found in, e.g. (Lin and Hsieh 1991; Plummer and Vaughan 1997; Kubalcik and Bobal 2006; Wei et al. 2010).

3.1 Multivariable system representation

The system model considered throughout this Chapter is in a discrete-time, multi-input multi-output (MIMO) transfer function form (Albertos and Sala 2004)

$$\boldsymbol{y}_k = \boldsymbol{G}(z^{-1})\boldsymbol{u}_k \tag{3.1}$$

where

$$\boldsymbol{y}_k = [y_{1,k}\ y_{2,k}\ \ldots\ y_{n,k}]^T \quad \text{and} \quad \boldsymbol{u}_k = [u_{1,k}\ u_{2,k}\ \ldots\ u_{n,k}]^T \tag{3.2}$$

denote the vectors containing the n system outputs and the n system inputs, respectively. Also,

$$\boldsymbol{G}(z^{-1}) = \begin{bmatrix} g_{11}(z^{-1}) & \cdots & g_{1n}(z^{-1}) \\ \vdots & \vdots & \vdots \\ g_{n1}(z^{-1}) & \cdots & g_{nn}(z^{-1}) \end{bmatrix} \tag{3.3}$$

denotes a square matrix containing individual transfer functions

$$g_{ij}(z^{-1}) = \frac{b_{1ij}z^{-1} + b_{2ij}z^{-2} + \ldots + b_{n_{bij}ij}z^{-n_{bij}}}{1 + a_{1ij}z^{-1} + a_{2ij}z^{-2} + \ldots + a_{n_{aij}ij}z^{-n_{aij}}} \tag{3.4}$$

with $i, j = 1, 2, \ldots, n$. Moreover, assume that the numerator and denominator polynomials in (3.4) are coprime, (3.4) is controllable and that $\boldsymbol{G}(z^{-1})$ is of full rank.

Remark 3.1.1. *Without loss of generality, in order to incorporate time delays, the corresponding leading numerator parameters in* (3.4) *are set to zero.*

3.1.1 Left matrix fraction description (LMFD)

The transfer function matrix representing the MIMO system (3.3) can be formulated in a left matrix fraction description (LMFD) (Kailath 1980)

$$\boldsymbol{G}(z^{-1}) = \boldsymbol{A}^{-1}(z^{-1})\boldsymbol{B}(z^{-1}) \tag{3.5}$$

so that the system representation (3.1) becomes

$$\boldsymbol{A}(z^{-1})\boldsymbol{y}_k = \boldsymbol{B}(z^{-1})\boldsymbol{u}_k \tag{3.6}$$

with

$$\begin{aligned}\boldsymbol{A}(z^{-1}) &= \boldsymbol{I}_n + \boldsymbol{A}_1 z^{-1} + \boldsymbol{A}_2 z^{-2} + \ldots + \boldsymbol{A}_{n_a} z^{-n_a} \\ \boldsymbol{B}(z^{-1}) &= \boldsymbol{B}_1 z^{-1} + \boldsymbol{B}_2 z^{-2} + \ldots + \boldsymbol{B}_{n_b} z^{-n_b}\end{aligned} \tag{3.7}$$

where $\boldsymbol{I}_n$ denotes the $n \times n$ identity and $\boldsymbol{A}_i \in \mathbb{R}^{n \times n}$ denote diagonal matrices containing the corresponding system model parameters. Similarly, $\boldsymbol{B}_j \in \mathbb{R}^{n \times n}$ denote matrices of corresponding system model parameters.

3.1.2 System diagonalisation

Considering the system representation (3.6) – (3.7), it is observed that the cross-coupling effects arise from the non-zero off-diagonal elements in $\boldsymbol{B}(z^{-1})$. In order to obtain a diagonal matrix (Plummer and Vaughan 1997; Kubalcik and Bobal 2006) proposed to include a cross-coupling compensation matrix which diagonalises the system by, effectively, transforming the system input

$$\boldsymbol{u}_k = \boldsymbol{E}(z^{-1})\boldsymbol{v}_k \tag{3.8}$$

with $\boldsymbol{v}_k = [v_{1,k}\ v_{2,k} \ldots\ v_{n,k}]^T$ and $\boldsymbol{E}(z^{-1})$ being defined to be

$$\boldsymbol{E}(z^{-1}) = adj[\boldsymbol{B}(z^{-1})]z^{\kappa} \tag{3.9}$$

the forward time shift z^{κ} is chosen such that $\boldsymbol{E}(z^{-1})$ just remains causal, see e.g. (Oppenheim et al. 1998), i.e. $\boldsymbol{E}(z^{-1})$ can be written as

$$\boldsymbol{E}(z^{-1}) = \boldsymbol{E}_0 + \boldsymbol{E}_1 z^{-1} + \boldsymbol{E}_2 z^{-2} + \ldots + \boldsymbol{E}_{n_e} z^{-n_e} \tag{3.10}$$

where $\boldsymbol{E}_i \in \mathbb{R}^{n\times n}$ denote matrices of corresponding model parameters. When substituting (3.8) into (3.6) yields

$$\boldsymbol{y}_k = \boldsymbol{A}^{-1}(z^{-1})\boldsymbol{B}_d(z^{-1})\boldsymbol{v}_k \tag{3.11}$$

with

$$\begin{aligned}\boldsymbol{B}_d(z^{-1}) &= \boldsymbol{B}(z^{-1})\boldsymbol{E}(z^{-1}) = det[\boldsymbol{B}(z^{-1})]z^{\kappa}\boldsymbol{I}_n \\ &= \boldsymbol{B}_{d,1}z^{-1} + \boldsymbol{B}_{d,2}z^{-2} + \ldots + \boldsymbol{B}_{d,n_{bd}}z^{-n_{bd}}\end{aligned} \tag{3.12}$$

being a diagonal matrix, and a diagonalised system representation (3.11) from the transformed or artificial input $\boldsymbol{v}_k$ to the output $\boldsymbol{y}_k$ is obtained.

3.2 Decoupling non-minimal state space MPC

The general non-minimal state-space system model, upon which the MPC controllers in their respective representations are based, can be straightforwardly obtained from the diagonalised system in the LMFD representation (3.11), as follows

$$\begin{aligned}\boldsymbol{x}_{g,k} &= \boldsymbol{G}_g\boldsymbol{x}_{g,k-1} + \boldsymbol{B}_g\boldsymbol{v}_{k-1} \\ \boldsymbol{y}_k &= \boldsymbol{C}_g\boldsymbol{x}_{g,k}\end{aligned} \tag{3.13}$$

with

$$\boldsymbol{G}_g = \begin{bmatrix} -\boldsymbol{A}_1 & -\boldsymbol{A}_2 & \cdots & -\boldsymbol{A}_{n_a-1} & -\boldsymbol{A}_{n_a} & \boldsymbol{B}_{d,2} & \cdots & \boldsymbol{B}_{d,n_{bd}-1} & \boldsymbol{B}_{d,n_{bd}} \\ \boldsymbol{I}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n \\ \boldsymbol{0}_n & \boldsymbol{I}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{I}_n & \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n \\ \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n \\ \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n & \boldsymbol{I}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n & \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{I}_n & \boldsymbol{0}_n \end{bmatrix} \tag{3.14}$$

and

$$\begin{aligned} \boldsymbol{B}_g &= [\boldsymbol{B}_{d,1}\ \boldsymbol{0}_n\ \boldsymbol{0}_n\ \cdots\ \boldsymbol{0}_n\ \boldsymbol{I}_n\ \boldsymbol{0}_n \cdots\ \boldsymbol{0}_n]^T \\ \boldsymbol{C}_g &= [\boldsymbol{I}_n\ \boldsymbol{0}_n\ \cdots\ \boldsymbol{0}_n] \end{aligned} \tag{3.15}$$

where $\boldsymbol{0}_n$ denotes a $n \times n$ matrix of zeros. The state vector is given by

$$\boldsymbol{x}_{g,k} = [\boldsymbol{y}_k^T\ \boldsymbol{y}_{k-1}^T\ \cdots\ \boldsymbol{y}_{k-n_a+1}^T\ \boldsymbol{v}_{k-1}^T\ \cdots\ \boldsymbol{v}_{k-n_{bd}+1}^T]^T \tag{3.16}$$

3.2.1 Incremental input form

As presented in (Wang and Young 2006; Wang 2009), the non-minimal state-space system representation in incremental input form is given by

$$\begin{aligned} \boldsymbol{x}_{\Delta,k} &= \boldsymbol{G}_\Delta \boldsymbol{x}_{\Delta,k-1} + \boldsymbol{B}_\Delta \Delta \boldsymbol{v}_{k-1} \\ \boldsymbol{y}_k &= \boldsymbol{C}_\Delta \boldsymbol{x}_{\Delta,k} \end{aligned} \tag{3.17}$$

with

$$\boldsymbol{G}_\Delta = \begin{bmatrix} \boldsymbol{G}_g & \boldsymbol{0}_{n_g \times n} \\ \boldsymbol{C}_g \boldsymbol{G}_g & \boldsymbol{I}_n \end{bmatrix} \quad \boldsymbol{B}_\Delta = \begin{bmatrix} \boldsymbol{B}_g \\ \boldsymbol{C}_g \boldsymbol{B}_g \end{bmatrix} \quad \boldsymbol{C}_\Delta = [\boldsymbol{0}_n \cdots \boldsymbol{0}_n \boldsymbol{I}_n] \tag{3.18}$$

and

$$\boldsymbol{x}_{\Delta,k}^T = [\Delta \boldsymbol{x}_{g,k}^T \ \boldsymbol{y}_k^T] \tag{3.19}$$

A modification to the usual linear state-space MPC formulation, see e.g. (Ikonen and Najim 2002; Kwon and Han 2005; Camacho and Bordons 2007; Wang 2009), is proposed here, that allows to assign individual prediction and control horizons, denoted Np and Nc, respectively, to the n input-output pairs $(\Delta v_i, y_i)$ $i = 1, 2, \ldots, n$. The vectors of the predicted system outputs $\boldsymbol{Y}$, future input differences $\boldsymbol{\Delta V}$ and the future reference trajectory $\boldsymbol{R}$ are defined to be

$$\begin{aligned} \boldsymbol{Y}_i &= [y_{i,k+1|k} \ y_{i,k+2|k} \ \cdots \ y_{i,k+Np_i|k}]^T \\ \boldsymbol{Y} &= [\boldsymbol{Y}_1^T \ \boldsymbol{Y}_2^T \ \cdots \ \boldsymbol{Y}_n^T]^T \end{aligned} \tag{3.20a}$$

$$\begin{aligned} \boldsymbol{\Delta V}_i &= [\Delta v_{i,k|k} \ \Delta v_{i,k+1|k} \ \cdots \ \Delta v_{i,k+Nc_i-1|k}]^T \\ \boldsymbol{\Delta V} &= [\boldsymbol{\Delta V}_1^T \ \boldsymbol{\Delta V}_2^T \ \cdots \ \boldsymbol{\Delta V}_n^T]^T \end{aligned} \tag{3.20b}$$

so that

$$\begin{aligned} \boldsymbol{V}_i &= [v_{i,k|k} \ v_{i,k+1|k} \ \cdots \ v_{i,k+Nc_i-1|k}]^T \\ \boldsymbol{V} &= [\boldsymbol{V}_1^T \ \boldsymbol{V}_2^T \ \cdots \ \boldsymbol{V}_n^T]^T \end{aligned} \tag{3.20c}$$

and

$$\begin{aligned} \boldsymbol{R}_i &= [r_{i,k+1|k} \ r_{i,k+2|k} \ \cdots \ r_{i,k+Np_i|k}]^T \\ \boldsymbol{R} &= [\boldsymbol{R}_1^T \ \boldsymbol{R}_2^T \ \cdots \ \boldsymbol{R}_n^T]^T \end{aligned} \tag{3.20d}$$

respectively. For completeness,

$$\begin{aligned} \boldsymbol{\Delta U}_i &= [\Delta u_{i,k|k}\ \Delta u_{i,k+1|k}\ \cdots\ \Delta u_{i,k+Nc_i-1|k}]^T \\ \boldsymbol{\Delta U\psi} &= [\boldsymbol{\Delta U}_1^T\ \boldsymbol{\Delta U}_2^T\ \cdots\ \boldsymbol{\Delta U}_n^T]^T \end{aligned} \tag{3.20e}$$

and

$$\begin{aligned} \boldsymbol{U}_i &= [u_{i,k|k}\ u_{i,k+1|k}\ \cdots\ u_{i,k+Nc_i-1|k}]^T \\ \boldsymbol{U\psi} &= [\boldsymbol{U}_1^T\ \boldsymbol{U}_2^T\ \cdots\ \boldsymbol{U}_n^T]^T \end{aligned} \tag{3.20f}$$

denote the current and predicted incremental inputs as well as the current and predicted inputs, respectively.

The cost function required to be minimised, however, is of the same form as in (Wang and Young 2006)

$$J_\Delta = (\boldsymbol{Y\psi} - \boldsymbol{R})^T \boldsymbol{Q}(\boldsymbol{Y\psi} - \boldsymbol{R}) + \boldsymbol{\Delta V}^T \boldsymbol{\Lambda} \boldsymbol{\Delta V\psi} \tag{3.21}$$

where $\boldsymbol{Q\psi} = diag(\boldsymbol{Q}_1 \boldsymbol{Q}_2 \ldots \boldsymbol{Q}_n)$ and $\boldsymbol{\Lambda} = diag(\boldsymbol{\Lambda}_1 \boldsymbol{\Lambda}_2 \ldots \boldsymbol{\Lambda}_n)$ are positive definite and positive semi-definite block diagonal weighting matrices, respectively, where the individual matrix blocks $\boldsymbol{Q}_i \in \mathbb{R}^{Np_i \times Np_i}$ and $\boldsymbol{\Lambda}_i \in \mathbb{R}^{Nc_i \times Nc_i}$ are themselves diagonal matrices.

In order to minimise the cost function (3.21) w.r.t the decision variables $\boldsymbol{\Delta V}$, i.e. input differences, requires that the output predictions $\boldsymbol{Y\psi}$ are expressed in terms of $\boldsymbol{\Delta V}$. As an exemplary, yet representative case, the output predictions

of the *ith* output, by taking (3.17) into account, are given by

$$
\begin{aligned}
y_{i,k+1|k} &= \boldsymbol{C}_{\Delta,i}\boldsymbol{x}_{\Delta,k+1|k} = \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}\boldsymbol{x}_{\Delta,k|k} + \boldsymbol{C}_{\Delta,i}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k} \\
y_{i,k+2|k} &= \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}^{2}\boldsymbol{x}_{\Delta,k|k} + \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k|k} \\
&\quad + \boldsymbol{C}_{\Delta,i}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k+1|k} \\
y_{i,k+3|k} &= \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}^{3}\boldsymbol{x}_{\Delta,k|k} + \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}^{2}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k|k} \\
&\quad + \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k+1|k} + \boldsymbol{C}_{\Delta,i}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k+2|k} \\
&\quad \vdots \\
y_{i,k+Nc_i|k} &= \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}^{Nc_i}\boldsymbol{x}_{\Delta,k|k} \\
&\quad + \boldsymbol{C}_{\Delta,i}\sum_{j=0}^{Nc_i-1}\Big(\boldsymbol{G}_{\Delta}^{j}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k+Nc_i-1-j|k} \\
&\quad \vdots \\
y_{i,k+Np_i|k} &= \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_{\Delta}^{Np_i}\boldsymbol{x}_{\Delta,k|k} \\
&\quad + \boldsymbol{C}_{\Delta,i}\sum_{j=Np_i-1}^{Np_i-Nc_i}\boldsymbol{G}_{\Delta}^{j}\boldsymbol{B}_{\Delta,i}\Delta v_{i,k+Np_i-1-j|k}
\end{aligned}
\tag{3.22}
$$

where $\boldsymbol{C}_{\Delta,i}$ denotes the *ith* row of $\boldsymbol{C}_{\Delta}$ and $\boldsymbol{B}_{\Delta,i}$ denotes the *ith* column of $\boldsymbol{B}_{\Delta}$, respectively. Furthermore, (3.22) can be written in a more compact form

$$
\boldsymbol{Y}_i = \boldsymbol{F}_i\boldsymbol{x}_{\Delta,k} + \boldsymbol{\Phi}_i\boldsymbol{\Delta V}_i \tag{3.23}
$$

with

$$\boldsymbol{F}_i = \begin{bmatrix} \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^2 \\ \vdots \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Nc_i} \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Nc_i+1} \\ \vdots \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Np_i} \end{bmatrix} \tag{3.24}$$

and

$$\boldsymbol{\Phi}_i = \begin{bmatrix} \boldsymbol{C}_{\Delta,i}\boldsymbol{B}_{\Delta,i} & 0 & \cdots & 0 \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta\boldsymbol{B}_{\Delta,i} & \boldsymbol{C}_{\Delta,i}\boldsymbol{B}_{\Delta,i} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Nc_i-1}\boldsymbol{B}_{\Delta,i} & \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Nc_i-2}\boldsymbol{B}_{\Delta,i} & \cdots & \boldsymbol{C}_{\Delta,i}\boldsymbol{B}_{\Delta,i} \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Nc_i}\boldsymbol{B}_{\Delta,i} & \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Nc_i-1}\boldsymbol{B}_{\Delta,i} & \cdots & \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta\boldsymbol{B}_{\Delta,i} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Np_i-1}\boldsymbol{B}_{\Delta,i} & \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Np_i-2}\boldsymbol{B}_{\Delta,i} & \cdots & \boldsymbol{C}_{\Delta,i}\boldsymbol{G}_\Delta^{Np_i-Nc_i}\boldsymbol{B}_{\Delta,i} \end{bmatrix} \tag{3.25}$$

Subsequently, the predictions of the n outputs are obtained from

$$\boldsymbol{Y\psi} = \boldsymbol{F}\boldsymbol{x}_{\Delta,k} + \boldsymbol{\Phi}\boldsymbol{\Delta}\boldsymbol{V\psi} \tag{3.26}$$

with $\boldsymbol{F\psi} = [\boldsymbol{F}_1^T \ \boldsymbol{F}_2^T \cdots \boldsymbol{F}_n^T]^T \in \mathbb{R}^{\sum_{i=1}^n Np_i \times n(n_a+n_{bd})}$ and $\boldsymbol{\Phi} = diag(\boldsymbol{\Phi}_1\boldsymbol{\Phi}_2\cdots\boldsymbol{\Phi}_n) \in \mathbb{R}^{\sum_{i=1}^n Np_i \times \sum_{i=1}^n Nc_i}$ being a block diagonal matrix.

Substituting (3.26) into the cost function (3.21) and solving the optimisation problem

$$\underset{\Delta\boldsymbol{V}}{min.} \quad J_\Delta \tag{3.27}$$

the incremental artificial input predictions are obtained in the usual state variable feedback form, cf. (Wang 2009)

$$\boldsymbol{\Delta V\psi} = -\boldsymbol{K}_{x_\Delta}\boldsymbol{x}_{\Delta,k} + \boldsymbol{K}_{R_\Delta}\boldsymbol{R\psi} \tag{3.28a}$$

where

$$\boldsymbol{K}_{x_\Delta} = (\boldsymbol{\Phi}^T\boldsymbol{Q\Phi} + \boldsymbol{\Lambda})^{-1}\boldsymbol{\Phi}^T\boldsymbol{QF\psi} \tag{3.28b}$$

$$\boldsymbol{K}_{R_\Delta} = (\boldsymbol{\Phi}^T\boldsymbol{Q\Phi} + \boldsymbol{\Lambda})^{-1}\boldsymbol{\Phi}^T\boldsymbol{Q\psi} \tag{3.28c}$$

denote the $\sum_{i=1}^{n} Nc_i \times n(n_a + n_{bd})$ and $\sum_{i=1}^{n} Nc_i \times \sum_{i=1}^{n} Np_i$ feedback gain matrices, respectively.

System input recovery

Solving the optimisation problem (3.27) results in obtaining the transformed system input (3.28a), however, the system input $\boldsymbol{u}_k$ is required to be obtained since this input is applied to the actual system.

Consider the input transformation (3.8) and associated transformation matrix $\boldsymbol{E}(z^{-1})$ defined in (3.9) and (3.10), respectively, which can also be written as

$$\boldsymbol{E}(z^{-1}) = \begin{bmatrix} e_{11}(z^{-1}) & \cdots & e_{1n}(z^{-1}) \\ \vdots & \vdots & \vdots \\ e_{n1}(z^{-1}) & \cdots & e_{nn}(z^{-1}) \end{bmatrix} \tag{3.29}$$

where

$$e_{jl}(z^{-1}) = e_{jl_0} + e_{jl_1}z^{-1} + e_{jl_2}z^{-2} + \ldots + e_{jl_{n_e}}z^{-n_e} \tag{3.30}$$

$\forall j, l = 1, 2, \ldots, n$. Moreover, without loss of generality and for the sake of simplicity, let the order of all the polynomials (3.30) identically be n_e, which also follows directly from (3.10). This may mean that some of the coefficients e_{jl_i} are

set to zero.

Remark 3.2.1. *In the case of MIMO systems with multiple time delays, i.e. the time delay of the individual transfer functions in* (3.3) *differ, some leading coefficients in* (3.30) *are set zero anyway.*

So, the *ith* input can be obtained as

$$u_{i,k} = \sum_{l=1}^{n} e_{il}(z^{-1}) v_{l,k} \tag{3.31}$$

and, subsequently, the predictions of the *ith* input are given by

$$u_{i,k+j} = \sum_{l=1}^{n} e_{il}(z^{-1}) v_{l,k+j} \qquad \forall j = 1, 2, \ldots, Nc_i - 1 \tag{3.32}$$

and, as a consequence, the entire vector containing current and predicted input values (3.20f) can be recovered by

$$\boldsymbol{U\psi} = \tilde{\boldsymbol{E}}\boldsymbol{V\psi} + \hat{\boldsymbol{E}}\hat{\boldsymbol{V}}\boldsymbol{\psi} \tag{3.33}$$

where

$$\tilde{\boldsymbol{E}} = \begin{bmatrix} \tilde{\boldsymbol{E}}_{11} & \cdots & \tilde{\boldsymbol{E}}_{1n} \\ \vdots & \vdots & \vdots \\ \tilde{\boldsymbol{E}}_{n1} & \cdots & \tilde{\boldsymbol{E}}_{nn} \end{bmatrix} \text{ with } \tilde{\boldsymbol{E}}_{jl} = \begin{bmatrix} e_{jl_0} & 0 & \cdots & 0 & 0 \\ e_{jl_1} & e_{jl_0} & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & e_{jl_0} & 0 \\ 0 & 0 & \cdots & e_{jl_1} & e_{jl_0} \end{bmatrix} \in \mathbb{R}^{Nc_j \times Nc_l} \tag{3.34}$$

and

$$\hat{\boldsymbol{E}} = \begin{bmatrix} \hat{\boldsymbol{E}}_1^T & \hat{\boldsymbol{E}}_2^T & \ldots & \hat{\boldsymbol{E}}_n^T \end{bmatrix}^T \tag{3.35a}$$

with

$$\hat{\boldsymbol{E}}_j = \begin{bmatrix} \begin{pmatrix} e_{j1_1} & e_{j1_2} & \cdots & e_{j1_{n_e-1}} & e_{j1_{n_e}} & e_{j2_1} & e_{j2_2} & \cdots & e_{j2_{n_e-1}} & e_{j2_{n_e}} \\ e_{j1_2} & e_{j1_3} & \cdots & e_{j1_{n_e}} & 0 & e_{j2_2} & e_{j2_3} & \cdots & e_{j2_{n_e}} & 0 \\ e_{j1_3} & e_{j1_4} & \cdots & 0 & 0 & e_{j2_3} & e_{j2_4} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ e_{j1_{n_e}} & 0 & \cdots & 0 & 0 & e_{j2_{n_e}} & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \end{bmatrix} \begin{pmatrix} \cdots & e_{jn_1} & e_{jn_2} & \cdots & e_{jn_{n_e-1}} & e_{jn_{n_e}} \\ \cdots & e_{jn_2} & e_{jn_3} & \cdots & e_{jn_{n_e}} & 0 \\ \cdots & e_{jn_3} & e_{jn_4} & \cdots & 0 & 0 \\ \cdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \cdots & e_{jn_{n_e}} & 0 & \cdots & 0 & 0 \\ \cdots & 0 & 0 & \cdots & 0 & 0 \\ \cdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \end{bmatrix} \in \mathbb{R}^{\sum_{i=1}^{n} Nc_i \times nn_e} \quad (3.35b)$$

Essentially, (3.33) is the representation of (3.31) and (3.32) in a compact matrix form. Also, note that (3.33) consists of a term corresponding to the current and predicted inputs and a term corresponding to previous inputs only, since

$$\hat{\boldsymbol{V}} = [v_{1,k-1}\ v_{1,k-2}\ \ldots\ v_{1,k-n_e}\ v_{2,k-1}\ v_{2,k-2}\ \ldots\ v_{2,k-n_e} \ldots\ v_{n,k-1}\ v_{n,k-2}\ \ldots\ v_{n,k-n_e+1}\ v_{n,k-n_e}]^T \quad (3.36)$$

Handling constraints

When imposing constraints on the system, it is desired that these do not introduce cross-couplings of the outputs. Consequently, if there is a change of the reference signal affecting, e.g. $\boldsymbol{Y}_1$, then, the remaining outputs are not supposed to be affected, despite imposed constraints on the input and/or output. The proposed modification of the NMSS MPC formulation in incremental input form allows one to consider the input-output pairs ($\boldsymbol{\Delta V}_i$, $\boldsymbol{Y}_i$) individually. This, in turn, allows the choice of which outputs are allowed to change and which are not by making use of the reference governor approach, see e.g. (Bemporad and Mosca 1994; Gilbert and Kolmanovsky 1995; Bemporad et al. 1997; Angeli and Mosca 1999; Exadaktylos et al. 2008). Similar to the closed-loop paradigm (CLP) (Rossiter 2004), where a perturbation term is added to the optimal, i.e. unconstrained, control law such that the constraints are fulfilled, the reference governor adapts the reference signal in order to avoid constraint violation, i.e.

$$\boldsymbol{W\psi} = \boldsymbol{R} + \boldsymbol{\Gamma} \tag{3.37}$$

where $\boldsymbol{W\psi} \in \mathbb{R}^{\sum_{i=1}^{n} Np_i \times 1}$ denotes the adapted reference signal and

$$\boldsymbol{\Gamma} = [\boldsymbol{\Gamma}_1^T \ \boldsymbol{\Gamma}_2^T \ \dots \ \boldsymbol{\Gamma}_n^T]^T \tag{3.38}$$

denotes the reference signal perturbation vector where

$$\boldsymbol{\Gamma}_i = [\gamma_{i,k+1|k} \ \gamma_{i,k+2|k} \ \dots \ \gamma_{i,k+Np_i|k}]^T \tag{3.39}$$

denotes the reference signal perturbation corresponding to the *ith* system output reference trajectory.

Since $\boldsymbol{\Gamma}$ describes a deviation from the desired reference trajectory, the values of its elements $\gamma_{i,k+j|k}$ are ideally zero, which corresponds to the case of inactive constraints. Moreover, in order to maintain offset-free steady-state set-point tracking, the sequences $\{\gamma_{i,k+1|k}\ \gamma_{i,k+2|k}\ \dots\ \gamma_{i,k+Np_i|k}\}\ \forall i = 1, 2, \dots, n$ are required to converge to zero, which is assumed here. However, to keep the deviation as small as possible, the following quadratic optimisation problem can be formulated

$$\begin{aligned} \underset{\boldsymbol{\Gamma}}{min.} \quad & \boldsymbol{\Gamma}^T\ \boldsymbol{\Gamma} \\ s.\ t. \quad & \boldsymbol{M\Gamma} \leq \boldsymbol{N\psi} \end{aligned} \tag{3.40}$$

where the constraints are required to be fulfilled element wise. The matrices $\boldsymbol{M\psi}$ and $\boldsymbol{N\psi}$of dimension $(4\sum_{i=1}^{n} Nc_i + 2\sum_{i=1}^{n} Np_i) \times \sum_{i=1}^{n} Np_i$ and $(4\sum_{i=1}^{n} Nc_i + 2\sum_{i=1}^{n} Np_i) \times 1$, respectively, formulate the constraints on the incremental system input, system input as well as system output, i.e.

$$\boldsymbol{M\psi} = \begin{bmatrix} \boldsymbol{M}_{\Delta U} \\ \boldsymbol{M}_U \\ \boldsymbol{M}_Y \end{bmatrix} \Bigg(\quad \text{and} \quad \boldsymbol{N\psi} = \begin{bmatrix} \boldsymbol{N}_{\Delta U} \\ \boldsymbol{N}_U \\ \boldsymbol{N}_Y \end{bmatrix} \Bigg(\tag{3.41}$$

respectively, and are subsequently derived.

At first, consider the case of imposing constraints on the incremental inputs $\boldsymbol{\Delta U}$, i.e.

$$\underline{\boldsymbol{\Delta U}}\boldsymbol{\psi} \leq \boldsymbol{\Delta U\psi} \leq \overline{\boldsymbol{\Delta U}}\boldsymbol{\psi} \tag{3.42}$$

where $\underline{(\cdot)}$ and $\overline{(\cdot)}$ denote the lower and upper boundaries, respectively. The optimal predicted control sequence (3.28), when considering the adapted reference signal $\boldsymbol{W}$, becomes

$$\boldsymbol{\Delta V\psi} = -\boldsymbol{K}_x \boldsymbol{x}_{\Delta,k} + \boldsymbol{K}_R(\boldsymbol{R} + \boldsymbol{\Gamma}) = \boldsymbol{\Delta V}_{opt} + \boldsymbol{K}_R\boldsymbol{\Gamma} \tag{3.43}$$

where the subscript $(\cdot)_{opt}$ denotes optimal, i.e. unconstrained, and is the solution of (3.21), i.e. (3.28). Substituting (3.43) into (3.33) after multiplying throughout with Δ, gives

$$\boldsymbol{\Delta U\psi} = \tilde{\boldsymbol{E}}\boldsymbol{\Delta V}_{opt} + \hat{\boldsymbol{E}}\boldsymbol{\Delta}\hat{\boldsymbol{V}} + \tilde{\boldsymbol{E}}\boldsymbol{K}_R\boldsymbol{\Gamma} = \boldsymbol{\Delta U}_{opt} + \tilde{\boldsymbol{E}}\boldsymbol{K}_R\boldsymbol{\Gamma} \tag{3.44}$$

so that the matrices formulating the constraints associated with $\boldsymbol{\Delta U}$, are given by

$$\boldsymbol{M}_{\Delta U} = \begin{bmatrix} -\tilde{\boldsymbol{E}}\boldsymbol{K}_R \\ \tilde{\boldsymbol{E}}\boldsymbol{K}_R \end{bmatrix}; \quad \boldsymbol{N}_{\Delta U} = \begin{bmatrix} \boldsymbol{\Delta U}_{opt} - \underline{\boldsymbol{\Delta U}}\boldsymbol{\psi} \\ \overline{\boldsymbol{\Delta U}} - \boldsymbol{\Delta U}_{opt} \end{bmatrix} \tag{3.45}$$

which are of dimension $2\sum_{i=1}^{n} Nc_i \times \sum_{i=1}^{n} Np_i$ and $2\sum_{i=1}^{n} Nc_i \times 1$, respectively.

Next, consider constraints on the input magnitude of $\boldsymbol{U}$, i.e.

$$\underline{\boldsymbol{U}}\boldsymbol{\psi} \leq \boldsymbol{U\psi} \leq \overline{\boldsymbol{U}}\boldsymbol{\psi} \tag{3.46}$$

these can be directly derived from above by expressing $\boldsymbol{\Delta U\psi}$ in a compact matrix form, i.e.

$$\boldsymbol{\Delta U\psi} = \boldsymbol{\Upsilon_1 U\psi} - \boldsymbol{\Upsilon_2}\ \boldsymbol{u}_{k-1} \tag{3.47}$$

where

$$\boldsymbol{\Upsilon}_1 = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -1 & 1 \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Upsilon}_2 = \begin{bmatrix} \left.\begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \end{matrix}\right\} Nc_1 \\ \left.\begin{matrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \end{matrix}\right\} Nc_2 \\ \vdots \\ \left.\begin{matrix} 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \end{matrix}\right\} Nc_n \end{bmatrix} \tag{3.48}$$

denote matrices of dimension $\sum_{i=1}^{n} Nc_i \times \sum_{i=1}^{n} Nc_i$ and $\sum_{i=1}^{n} Nc_i \times n$, respectively. So, the constraints can be formulated to be

$$\boldsymbol{M}_U = \begin{bmatrix} -\tilde{\boldsymbol{E}}\boldsymbol{K}_R \\ \tilde{\boldsymbol{E}}\boldsymbol{K}_R \end{bmatrix}; \quad \boldsymbol{N}_U = \begin{bmatrix} \boldsymbol{\Delta U}_{opt} - \boldsymbol{\Upsilon}_1 \underline{\boldsymbol{U}}\boldsymbol{\psi} + \boldsymbol{\Upsilon}_2 \boldsymbol{u}_{k-1} \\ \boldsymbol{\Upsilon}_1 \overline{\boldsymbol{U}}\boldsymbol{\psi} - \boldsymbol{\Delta U}_{opt} - \boldsymbol{\Upsilon}_2 \boldsymbol{u}_{k-1} \end{bmatrix} \tag{3.49}$$

and these matrices are of dimension $2\sum_{i=1}^{n} Nc_i \times \sum_{i=1}^{n} Np_i$ and $2\sum_{i=1}^{n} Nc_i \times 1$, respectively.

Finally, consider constraints on the outputs $\boldsymbol{Y}$, i.e.

$$\underline{\boldsymbol{Y}}\boldsymbol{\psi} \leq \boldsymbol{Y}\boldsymbol{\psi} \leq \overline{\boldsymbol{Y}}\boldsymbol{\psi} \tag{3.50}$$

Substituting (3.43) into the output prediction equation (3.26), the constraints

formulation is obtained as

$$\boldsymbol{M}_Y = \begin{bmatrix} -\boldsymbol{\Phi}\boldsymbol{K}_R \\ \boldsymbol{\Phi}\boldsymbol{K}_R \end{bmatrix}; \quad \boldsymbol{N}_Y = \begin{bmatrix} \boldsymbol{F}\boldsymbol{x}_{\Delta,k} + \boldsymbol{\Phi}\boldsymbol{\Delta}\boldsymbol{V}_{opt} - \underline{\boldsymbol{Y}}\boldsymbol{\psi} \\ \overline{\boldsymbol{Y}}\boldsymbol{\psi} - \boldsymbol{F}\boldsymbol{x}_{\Delta,k} - \boldsymbol{\Phi}\boldsymbol{\Delta}\boldsymbol{V}_{opt} \end{bmatrix} \tag{3.51}$$

where the dimensions of $\boldsymbol{M}_Y$ and $\boldsymbol{N}_Y$ are given by $2\sum_{i=1}^{n} Np_i \times \sum_{i=1}^{n} Np_i$ and $2\sum_{i=1}^{n} Np_i \times 1$, respectively.

Regarding the optimisation problem (3.40) and the subsequent derivation of the constraints formulation, the following observations can be made:

- When constraints are not violated by (3.28), then $\boldsymbol{\Gamma} = \boldsymbol{0}$, so that, from (3.40), $\boldsymbol{N}\boldsymbol{\psi} \geq \boldsymbol{0}$. This, in turn, means that (3.40) is required to be solved only if at least one element in the vector $\boldsymbol{N}\boldsymbol{\psi}$ is negative, i.e. $[\boldsymbol{N}]_j < 0 \ \forall j$, where $[\boldsymbol{N}]_j$ denotes the jth element of $\boldsymbol{N}$.

- In order to avoid cross-coupling effects introduced by the constraints, corresponding reference trajectory deviations can be chosen, e.g. $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}_1$, which are allowed to vary. This, effectively, forces the remaining reference trajectory deviations to be zero. Also, the matrices $\boldsymbol{M}\boldsymbol{\psi}$ and $\boldsymbol{N}\boldsymbol{\psi}$ can be truncated accordingly. This allows the dimension of the optimisation problem (3.40) to be kept low.

3.2.2 Integral-of-errors state variable formulation

The NMSS model with an integral-of-errors state variable (Young et al. 1987; Wang and Young 1988), based on the general NMSS representation (3.13)–(3.15) of the diagonalised system, is given by

$$\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{G}\boldsymbol{x}_{k-1} + \boldsymbol{B}\boldsymbol{v}_{k-1} + \boldsymbol{D}\boldsymbol{r}_k \\ \boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k \end{aligned} \tag{3.52}$$

with

$$\boldsymbol{x}_k = \begin{bmatrix} \boldsymbol{x}_{g,k} \\ \boldsymbol{\zeta}_k \end{bmatrix} \quad \boldsymbol{G} = \begin{bmatrix} \boldsymbol{G}_g & \boldsymbol{0}_{n_g \times n} \\ \boldsymbol{C}_g \boldsymbol{G}_g & \boldsymbol{I}_n \end{bmatrix} \quad \boldsymbol{B} = \begin{bmatrix} \boldsymbol{B}_g \\ \boldsymbol{C}_g \boldsymbol{B}_g \end{bmatrix} \tag{3.53}$$

where $\boldsymbol{\zeta}_k = \boldsymbol{\zeta}_{k-1} + \boldsymbol{r}_k - \boldsymbol{y}_k$ denotes the integral-of-errors state variable and

$$\boldsymbol{C} = [\boldsymbol{I}_n \ \boldsymbol{0}_n \dots \boldsymbol{0}_n] \qquad \boldsymbol{D} = [\boldsymbol{0}_n \ \dots \ \boldsymbol{0}_n \ \boldsymbol{I}_n]^T \tag{3.54}$$

In the MPC formulation based on the NMSS model that uses an integral-of-errors state variable, which ensures set-point tracking, instead of the system outputs, the state vector is predicted (Exadaktylos 2007, Chap. 4). Consequently, the cost function that is required to be minimized w.r.t $\bar{\boldsymbol{V}}$, is given by

$$J = \boldsymbol{X}^T \bar{\boldsymbol{Q}} \boldsymbol{X} + \bar{\boldsymbol{V}}^T \bar{\boldsymbol{\Lambda}} \bar{\boldsymbol{V}} \boldsymbol{\psi} \tag{3.55}$$

where

$$\boldsymbol{X} = [\boldsymbol{x}_{k+1|k}^T \ \boldsymbol{x}_{k+2|k}^T \ \dots \ \boldsymbol{x}_{k+N_p|k}^T]^T \tag{3.56}$$

denotes the $n(n_a + n_{bd})N_p \times 1$ dimensional vector of predicted states and

$$\bar{\boldsymbol{V}} = [\boldsymbol{v}_{k|k}^T \ \boldsymbol{v}_{k+1|k}^T \ \dots \ \boldsymbol{v}_{k+N_c-1|k}^T]^T \tag{3.57}$$

denotes the transformed input prediction vector of dimension $nN_c \times 1$. Additionally, $\bar{\boldsymbol{Q}} \in \mathbb{R}^{n(n_a+n_{bd})N_p \times n(n_a+n_{bd})N_p}$ and $\bar{\boldsymbol{\Lambda}} \in \mathbb{R}^{nN_c \times nN_c}$ denote positive definite and positive semi-definite weighting matrices, respectively.

In a similar manner as in (3.22) and, subsequently, in (3.23) the pre-

dicted states are obtained by

$$\boldsymbol{X\psi} = \bar{\boldsymbol{F}}\boldsymbol{x}(k) + \bar{\boldsymbol{\Phi}}\bar{\boldsymbol{V}}\boldsymbol{\psi} + \boldsymbol{H}\bar{\boldsymbol{R}}\boldsymbol{\psi} \tag{3.58}$$

where

$$\bar{\boldsymbol{F}}\boldsymbol{\psi} = [\boldsymbol{G}^T \ (\boldsymbol{G}^2)^T \ \dots \ (\boldsymbol{G}^{N_p})^T]^T \tag{3.59a}$$

$$\bar{\boldsymbol{\Phi}} = \begin{bmatrix} \begin{pmatrix} \boldsymbol{B}\boldsymbol{\psi} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{G}\boldsymbol{B}\boldsymbol{\psi} & \boldsymbol{B}\boldsymbol{\psi} & \cdots & \boldsymbol{0} & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \end{pmatrix} \\ \boldsymbol{G}^{N_c-2}\boldsymbol{B} \quad \boldsymbol{G}^{N_c-1}\boldsymbol{B}\boldsymbol{\psi} \cdots \quad \boldsymbol{B}\boldsymbol{\psi} \quad \boldsymbol{0} \\ \boldsymbol{G}^{N_c-1}\boldsymbol{B} \quad \boldsymbol{G}^{N_c-2}\boldsymbol{B}\boldsymbol{\psi} \cdots \quad \boldsymbol{G}\boldsymbol{B}\boldsymbol{\psi} \quad \boldsymbol{B}\boldsymbol{\psi} \\ \boldsymbol{G}^{N_c}\boldsymbol{B} \quad \boldsymbol{G}^{N_c-1}\boldsymbol{B}\boldsymbol{\psi} \cdots \quad \boldsymbol{G}^2\boldsymbol{B}\boldsymbol{\psi} \quad \boldsymbol{G}\boldsymbol{B} + \boldsymbol{B}\boldsymbol{\psi} \\ \boldsymbol{G}^{N_c+1}\boldsymbol{B}\boldsymbol{\psi} \quad \boldsymbol{G}^{N_c}\boldsymbol{B}\boldsymbol{\psi} \ \cdots \quad \boldsymbol{G}^3\boldsymbol{B} \quad \boldsymbol{G}^2\boldsymbol{B} + \boldsymbol{G}\boldsymbol{B} + \boldsymbol{B} \\ \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \\ \begin{pmatrix} \boldsymbol{G}^{N_p-1}\boldsymbol{B} & \boldsymbol{G}^{N_p-2}\boldsymbol{B} & \cdots & \boldsymbol{G}^{N_p-N_c+1}\boldsymbol{B}\boldsymbol{\psi} & \sum_{i=0}^{N_p-N_c}\boldsymbol{G}^i\boldsymbol{B} \end{pmatrix} \end{bmatrix} \begin{pmatrix} \\ \\ \\ \\ \\ \\ \\ \\ \end{pmatrix} \tag{3.59b}$$

$$\boldsymbol{H}\boldsymbol{\psi} = \begin{bmatrix} \boldsymbol{D}\boldsymbol{\psi} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{G}\boldsymbol{D}\boldsymbol{\psi} & \boldsymbol{D}\boldsymbol{\psi} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{G}^{N_p-1}\boldsymbol{D} & \boldsymbol{G}^{N_p-2}\boldsymbol{D} & \cdots & \boldsymbol{D}\boldsymbol{\psi} \end{bmatrix} \tag{3.59c}$$

and the vector of the future reference trajectories is defined to be

$$\bar{\boldsymbol{R}} = [\boldsymbol{r}_{k+1}^T \ \boldsymbol{r}_{k+2}^T \ \dots \ \boldsymbol{r}_{k+N_p}^T]^T \tag{3.60}$$

A detailed derivation and further information on the MPC based on NMSS models with an integral-of-errors state variable can be found in (Exadaktylos 2007, Chap. 4).

The (unconstrained/optimal) control law is obtained by solving the optimisation problem

$$\underset{\bar{\boldsymbol{V}}}{min.} \quad J \tag{3.61}$$

for which the solution is given by, similar to (3.28),

$$\bar{\boldsymbol{V}} = -\bar{\boldsymbol{K}}_x \boldsymbol{x}_k + \bar{\boldsymbol{K}}_R \bar{\boldsymbol{R}} \tag{3.62a}$$

with

$$\bar{\boldsymbol{K}}_x = (\bar{\boldsymbol{\Phi}}^T \bar{\boldsymbol{Q}} \bar{\boldsymbol{\Phi}} + \bar{\boldsymbol{\Lambda}})^{-1} \bar{\boldsymbol{\Phi}}^T \bar{\boldsymbol{Q}} \bar{\boldsymbol{F}} \tag{3.62b}$$

$$\bar{\boldsymbol{K}}_R = -(\bar{\boldsymbol{\Phi}}^T \bar{\boldsymbol{Q}} \bar{\boldsymbol{\Phi}} + \bar{\boldsymbol{\Lambda}})^{-1} \bar{\boldsymbol{\Phi}}^T \bar{\boldsymbol{Q}} \boldsymbol{H} \boldsymbol{\psi} \tag{3.62c}$$

Remark 3.2.2. *The cost function of the NMSS MPC in the integral-of-errors state variable representation* (3.55) *implicitly depends on the reference trajectories via the integral-of-errors state variable* $\boldsymbol{\zeta}_k$ *and its predictions, so that offset free steady-state tracking is achieved (Exadaktylos 2007, Chap 4).*

As a consequence, when making use of a cost function similar to (3.21)*, i.e.* $\bar{J} = (\boldsymbol{Y}\boldsymbol{\psi} - \boldsymbol{R})^T \boldsymbol{Q} (\boldsymbol{Y}\boldsymbol{\psi} - \boldsymbol{R}) + \boldsymbol{V}^T \boldsymbol{\Lambda} \boldsymbol{V}$*, instead of* (3.55) *applied to the NMSS MPC with integral-of-errors state variable via the relationship* $\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k$ *from* (3.52) *in order to obtain the required output predictions, results in nullifying the integral-of-errors state variables so that the performance is impaired and, consequently, offset free set-point tracking is not ensured.*

At this juncture, certain observations can be made regarding the NMSS MPC with integral-of-errors state variable compared to the NMSS MPC in incremental input form:

- Due to the different cost functions used (3.21) and (3.55), respectively, in particular that (3.21) depends explicitly on the system output predic-

tions while (3.55) depends on the state vector predictions (and implicitly on the system output predictions), it is straightforward to 'separate' the input-output pairs $(\boldsymbol{Y}_i, \boldsymbol{\Delta V}_i)$ and assign individual control and prediction horizons to each pair in the incremental input form, which can be used as additional tuning parameters. Other than in the integral-of-errors state variable form where the prediction horizon relates to the state vector and not to the system outputs. Consequently, there is a single prediction and control horizon only.

- Although both NMSS MPC representations are based on the same diagonalised system representation (3.13), which was achieved by a system input transformation (3.8), the matrices involved in recovering of the system input predictions (3.33), differ due to the definitions of the transformed input prediction vectors (3.20b) and (3.57), respectively.

System input recovery

As mentioned above, the system input $\boldsymbol{u}_k$ cannot be recovered using (3.33) – (3.36), however, since (3.8) also applies to the NMSS MPC in integral-of-errors state variable form, (3.8) written in matrix form, similar to (3.33), yields

$$\bar{\boldsymbol{U}} = \bar{\bar{\boldsymbol{E}}}\bar{\boldsymbol{V}} + \hat{\bar{\boldsymbol{E}}}\hat{\bar{\boldsymbol{V}}} \tag{3.63}$$

where

$$\bar{\boldsymbol{U}} = [\boldsymbol{u}_{k|k}^T \; \boldsymbol{u}_{k+1|k}^T \; \ldots \; \boldsymbol{u}_{k+N_c-1|k}^T]^T \tag{3.64}$$

denotes the $nN_c \times 1$ vector of the current and predicted system inputs. Taking (3.10) into account and similar to (3.34), $\bar{\bar{\boldsymbol{E}}} \in \mathbb{R}^{nN_c \times nN_c}$ is given by

$$\bar{\bar{\boldsymbol{E}}} = \begin{bmatrix} \boldsymbol{E}_0 & \mathbf{0}_n & \cdots & \mathbf{0}_n & \mathbf{0}_n \\ \boldsymbol{E}_1 & \boldsymbol{E}_0 & \cdots & \mathbf{0}_n & \mathbf{0}_n \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0}_n & \mathbf{0}_n & \cdots & \boldsymbol{E}_0 & \mathbf{0}_n \\ \mathbf{0}_n & \mathbf{0}_n & \cdots & \boldsymbol{E}_1 & \boldsymbol{E}_0 \end{bmatrix} \tag{3.65}$$

and similar to (3.35), $\bar{\bar{\hat{\boldsymbol{E}}}} \in \mathbb{R}^{nN_c \times nn_e}$ is given by

$$\bar{\bar{\hat{\boldsymbol{E}}}} = \begin{bmatrix} \boldsymbol{E}_1 & \boldsymbol{E}_2 & \cdots & \boldsymbol{E}_{n_e-1} & \boldsymbol{E}_{n_e} \\ \boldsymbol{E}_2 & \boldsymbol{E}_3 & \cdots & \boldsymbol{E}_{n_e} & \mathbf{0}_n \\ \boldsymbol{E}_3 & \boldsymbol{E}_4 & \cdots & \mathbf{0}_n & \mathbf{0}_n \\ \vdots & \vdots & \iddots & \vdots & \vdots \\ \boldsymbol{E}_{n_e} & \mathbf{0}_n & \cdots & \mathbf{0}_n & \mathbf{0}_n \\ \mathbf{0}_n & \mathbf{0}_n & \cdots & \mathbf{0}_n & \mathbf{0}_n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_n & \mathbf{0}_n & \cdots & \mathbf{0}_n & \mathbf{0}_n \end{bmatrix} \tag{3.66}$$

and

$$\bar{\bar{\hat{\boldsymbol{V}}}} = [\boldsymbol{v}_{k-1}^T \ \boldsymbol{v}_{k-2}^T \ \ldots \ \boldsymbol{v}_{k-n_e+1}^T \ \boldsymbol{v}_{k-n_e}^T]^T \tag{3.67}$$

Handling constraints

Since here, as mentioned above, the actual system outputs are not predicted but rather the state vector, the method of constraint handling via reference trajectory adaptation is not straightforwardly applicable. Moreover, as discussed above, since the performance of the system is greatly determined by the integral-of-errors

state variable, which includes the reference signal, as a consequence, reference trajectory adaptation impacts the performance significantly. Hence, the constraints are handled by using the 'conventional' method as proposed in (Exadaktylos 2007, Chap. 4), i.e. the quadratic optimisation problem

$$\begin{aligned} &\underset{\bar{\boldsymbol{V}}}{min.} \quad J \\ &s.\,t. \quad \bar{\boldsymbol{M}}\bar{\boldsymbol{V}}\boldsymbol{\psi} \leq \bar{\boldsymbol{N}} \end{aligned} \tag{3.68}$$

is required to be solved, where

$$\bar{\boldsymbol{M}} = \begin{bmatrix} -\bar{\bar{\boldsymbol{E}}}\boldsymbol{\psi} \\ \bar{\bar{\boldsymbol{E}}} \\ -\bar{\boldsymbol{\Upsilon}}_1\bar{\bar{\boldsymbol{E}}} \\ \bar{\boldsymbol{\Upsilon}}_1\bar{\bar{\boldsymbol{E}}}\boldsymbol{\psi} \\ -\bar{\boldsymbol{C}}\bar{\boldsymbol{\Phi}} \\ \bar{\boldsymbol{C}}\bar{\boldsymbol{\Phi}} \end{bmatrix} \quad \bar{\boldsymbol{N}} = \begin{bmatrix} -\underline{\bar{\boldsymbol{U}}} + \bar{\bar{\boldsymbol{E}}}\bar{\hat{\boldsymbol{V}}}\boldsymbol{\psi} \\ \overline{\bar{\boldsymbol{U}}} - \bar{\bar{\boldsymbol{E}}}\bar{\hat{\boldsymbol{V}}}\boldsymbol{\psi} \\ -\underline{\Delta\bar{\boldsymbol{U}}}\boldsymbol{\psi} + \bar{\boldsymbol{\Upsilon}}_1\bar{\bar{\boldsymbol{E}}}\bar{\hat{\boldsymbol{V}}}\boldsymbol{\psi} - \bar{\boldsymbol{\Upsilon}}_2\boldsymbol{u}_{k-1} \\ \overline{\Delta\bar{\boldsymbol{U}}}\boldsymbol{\psi} - \bar{\boldsymbol{\Upsilon}}_1\bar{\bar{\boldsymbol{E}}}\bar{\hat{\boldsymbol{V}}}\boldsymbol{\psi} + \bar{\boldsymbol{\Upsilon}}_2\boldsymbol{u}_{k-1} \\ -\underline{\bar{\boldsymbol{Y}}}\boldsymbol{\psi} + \bar{\boldsymbol{C}}\bar{\boldsymbol{F}}\boldsymbol{x}_k + \bar{\boldsymbol{C}}\boldsymbol{H}\bar{\boldsymbol{R}}\boldsymbol{\psi} \\ \overline{\bar{\boldsymbol{Y}}}\boldsymbol{\psi} - \bar{\boldsymbol{C}}\bar{\boldsymbol{F}}\boldsymbol{x}_k - \bar{\boldsymbol{C}}\boldsymbol{H}\bar{\boldsymbol{R}}\boldsymbol{\psi} \end{bmatrix} \tag{3.69a}$$

with

$$\bar{\boldsymbol{\Upsilon}}_1 = \begin{bmatrix} \boldsymbol{I}_n & \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n \\ -\boldsymbol{I}_n & \boldsymbol{I}_n & \boldsymbol{0}_n & \cdots & \boldsymbol{0}_n \\ \boldsymbol{0}_n & -\boldsymbol{I}_n & \boldsymbol{I}_n & \cdots & \boldsymbol{0}_n \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \boldsymbol{0}_n & \boldsymbol{0}_n & \cdots & -\boldsymbol{I}_n & \boldsymbol{I}_n \end{bmatrix} \in \mathbb{R}^{nN_c \times nN_c} \quad \bar{\boldsymbol{\Upsilon}}_2 = \begin{bmatrix} \boldsymbol{I}_n \\ \boldsymbol{0}_n \\ \vdots \\ \boldsymbol{0}_n \end{bmatrix} \in \mathbb{R}^{nN_c \times n} \tag{3.69b}$$

and $\bar{\boldsymbol{C}} = diag(\boldsymbol{C}, \boldsymbol{C}, \ldots, \boldsymbol{C}) \in \mathbb{R}^{nN_p \times (n_a + n_{bd})N_p}$ denotes a block diagonal matrix.

Handling the constraints in this way, clearly impairs the output decoupling properties, see Section 3.3.1, and, other than in the NMSS MPC in incremental input form that uses the reference trajectory adaptation method,

constraint violations cannot be detected in such a straightforward way, hence the optimisation problem (3.68) is required to be solved at every sampling instance.

3.3 Simulation example: Quadruple Tank Process

The quadruple tank process is a MIMO system comprising 2 inputs and 2 outputs, i.e. $n = 2$, and is developed by (Johansson 2000). In Figure 3.1, the schematic of the system is shown, from where the cross-coupling effects become apparent. The aim is to control the water levels of the lower tanks, where system output y_1 denotes the water level of tank 1 and system output y_2 denotes the water level of tank 2, respectively. The system inputs u_1 and u_2 denote the voltages applied to the electrically driven water pumps, which feed the tanks. Further information on the operational principle and its properties can be found in, e.g. (Johansson and Nunes 1998; Johansson 2000, 2002).

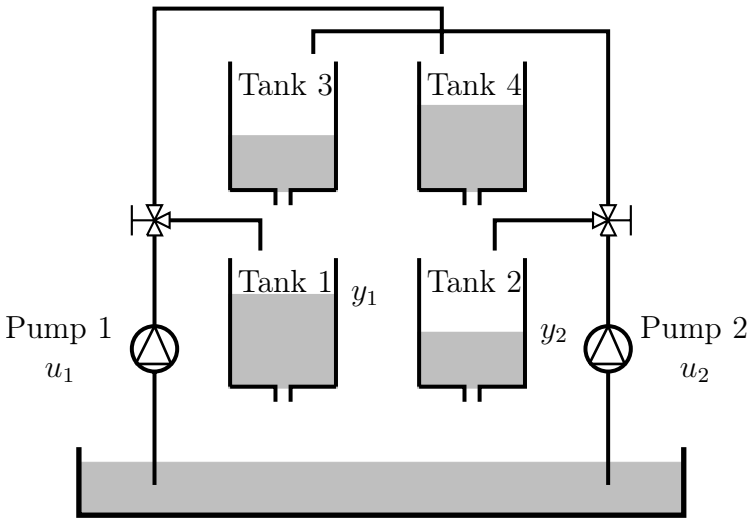


Figure 3.1: Schematic of the quadruple tank process

The minimum-phase system model in the continuous-time domain is adopted from (Johansson 2000) and discretized by making use of the Matlab®

function `c2d` where the sampling time is chosen to be $T_s = 5\,\mathrm{s}$. The discrete-time system model is obtained as

$$\boldsymbol{A}_1 = \begin{bmatrix} -1.7271 & 0 \\ 0 & -1.7924 \end{bmatrix} \quad \boldsymbol{A}_2 = \begin{bmatrix} 0.7423 & 0 \\ 0 & 0.8007 \end{bmatrix} \tag{3.70a}$$

and

$$\boldsymbol{B}_1 = \begin{bmatrix} 0.2014 & 0.0119 \\ 0.006 & 0.1513 \end{bmatrix} \quad \boldsymbol{B}_2 = \begin{bmatrix} -0.1621 & 0.0108 \\ 0.0056 & -0.1281 \end{bmatrix} \tag{3.70b}$$

Implementation results of the NMSS MPC in incremental input form incorporating the proposed modification, according to Section 3.2.1, are presented in Figure 3.2. Here, both the NMSS MPC in incremental input form using the diagonalised system model (solid line) according to Section 3.1.2, i.e. the decoupling technique, and using the non-diagonalised system model (3.70) directly (dashed line), are shown.

The reference signal is a step from 12 units to 14 units at sampling instance $k = 200$ for system output y_1 and at sampling instance $k = 700$ for system output y_2. The prediction and control horizons are chosen to be $Np_1 = Np_2 = 25$ and $Nc_1 = Nc_2 = 20$, respectively, for both MPC controller formulations. The weighting matrices are chosen to be identity matrices for the MPC based on the diagonalised model and the diagonal elements of the MPC based on the non-diagonalised model are chosen such that similar output responses are achieved.

It can be observed that, as expected, when the system model in diagonalised form is employed, the cross-coupling effects are eliminated (without model-mismatch). This is in contrast to the case where the non-diagonalised system model is used, where cross-coupling effects are clearly visible. Moreover, it is observed that the transformed system inputs v_1 and v_2 respond to the reference

signal changes corresponding to their system outputs only. This confirms the expectation that the input-output pairs, i.e. (v_1, y_1) and (v_2, y_2), can be considered individually. On the other hand, the system inputs u_1 and u_2 respond to the reference signal changes of both system outputs, which is to be expected in a system exhibiting cross-coupling effects.

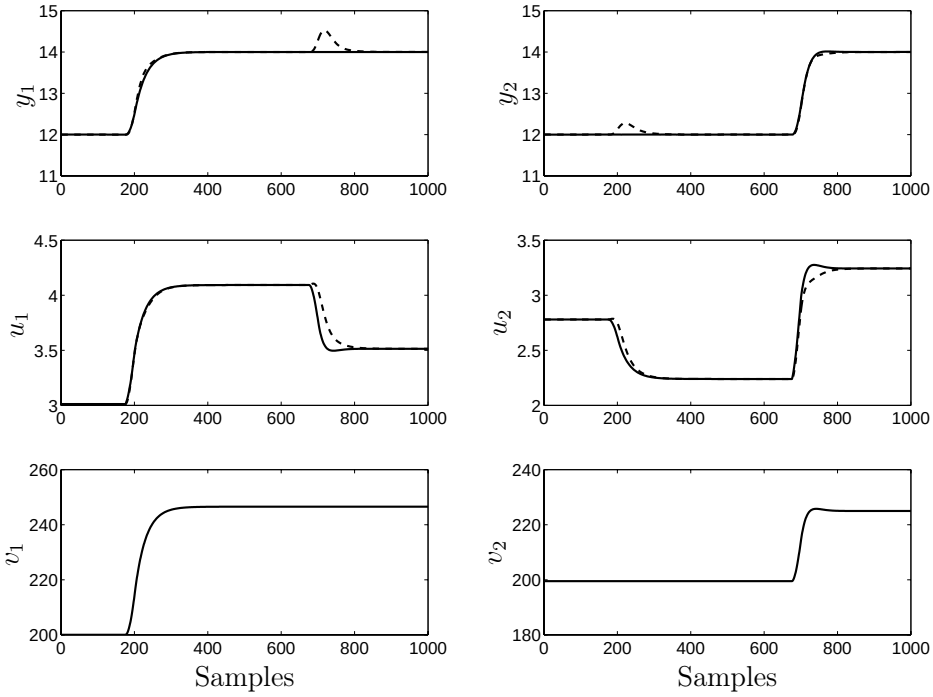


Figure 3.2: Implementation results of the NMSS MPC in incremental input form where the solid line corresponds to the proposed MPC formulation using the diagonalised system model and the dashed line corresponds to the MPC using the non-diagonalised system model. Upper: system outputs. Middle: system inputs and lower: transformed system input.

Similar to the above, implementation results of the NMSS MPC using an integral-of-errors state variable, as discussed in Section 3.2.2, are shown in Figure 3.3. Again, implementation results of the MPC based on the diagonalised (solid line) and non-diagonalised (dashed line) system model are presented. The reference signal applied to those for the incremental input form, as well as the

prediction and control horizon, are identical. Additionally, in the case of the MPC based on the diagonalised system model, the weighting matrices $\bar{\boldsymbol{Q}}_{\boldsymbol{\psi}}$ and $\bar{\boldsymbol{\Lambda}}$ are chosen to be identity matrices except that the weighting elements in $\bar{\boldsymbol{Q}}$, corresponding to the integral-of-errors state variable and respective predictions, are chosen to be [3 1.5]. In the case of using the non-diagonalised model, the weighting matrices are of a diagonal form and, again, tuned such that an (almost) identical output response is achieved.

When comparing Figure 3.2 and Figure 3.3, it can be observed that almost identical performance is achieved. Both NMSS MPC formulations, the incremental input form and integral-of-errors state variable form, are able to eliminate the cross-coupling effects, if the diagonalised model is used and no model mismatch is present.

Remark 3.3.1. *During various simulation examples of different systems, it could be observed that in the case of dealing with multiple time-delays, which significantly differ from each other, the performance of the NMSS MPC according to Section 3.2.1 as well as Section 3.2.2 is noticeably impaired.*

3.3.1 Imposing constraints

It is desired that, despite imposed constraints, the decoupling properties are preserved. As such, the NMSS MPC controllers based on the diagonalized system are considered here only. Also, the simulation set-up, i.e. reference signal, weighting matrices and horizons, is identical as used in the unconstrained scenarios as shown in Figures 3.2 and 3.3, respectively.

At first, consider the NMSS MPC in incremental input form, as discussed in Section 3.2.1, where use is made of the reference trajectory adaptation method (3.40) in order to handle constraints. The proposed modification of the

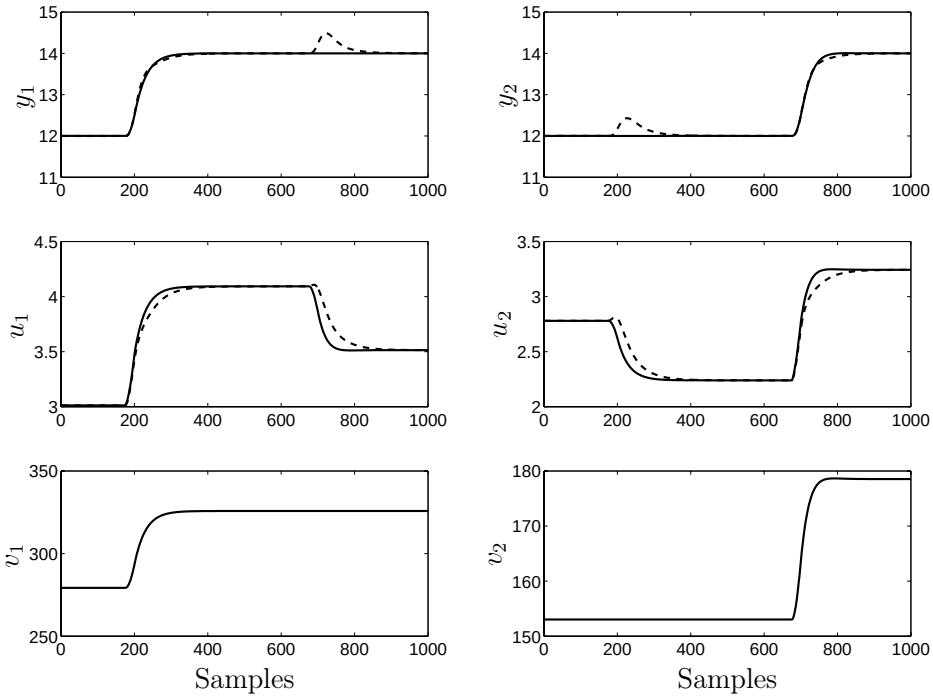


Figure 3.3: Implementation results of the NMSS MPC in integral-of-errors state variable form where the solid line corresponds to the MPC formulation using the diagonalised system model and the dashed line corresponds to the MPC using the non-diagonalised system model. Upper: system outputs. Middle: system inputs and lower: transformed system input.

MPC formulation (3.20) makes it straightforward to suppress the cross-coupling effects by allowing only the reference trajectory deviations, corresponding to the desired reference set-point change, to be non-zero. Also, as already mentioned in Section 3.2.1, the dimension of the optimisation problem (3.40) required to be actually solved, can be reduced accordingly. This is demonstrated in the following.

Consider constraints on the first incremental input, i.e. $-0.01 \leq \boldsymbol{\Delta U}_1 \leq 0.01$, so that $\underline{\boldsymbol{\Delta U}_1} = -0.01$ and $\overline{\boldsymbol{\Delta U}}_1 = 0.01$, respectively. Furthermore, two scenarios, which correspond to the set-point change of the output y_1 and the output y_2, respectively, are required to be considered. In the first scenario, $\boldsymbol{\Gamma}_1 \neq \mathbf{0}$

while $\boldsymbol{\Gamma}_2 = \mathbf{0}$, which accounts for a set-point change in y_1 while y_2 is supposed to be constant, hence the optimisation problem (3.40) becomes

$$\begin{aligned} \underset{\boldsymbol{\Gamma}_1}{min.} \quad & \boldsymbol{\Gamma}_1^T \boldsymbol{\Gamma}_1 \\ s.\,t. \quad & \boldsymbol{M}_{\Delta U_1}^{(1)} \boldsymbol{\Gamma}_1 \leq \boldsymbol{N}_{\Delta U_1} \end{aligned} \tag{3.71}$$

where

$$\boldsymbol{M}_{\Delta U_1}^{(1)} = \begin{bmatrix} \tilde{\boldsymbol{E}}_{U_1} \boldsymbol{K}_{R_1} \\ -\tilde{\boldsymbol{E}}_{U_1} \boldsymbol{K}_{R_1} \end{bmatrix} \in \mathbb{R}^{2Nc_1 \times Np_1} \quad \boldsymbol{N}_{\Delta U_1} = \begin{bmatrix} \boldsymbol{\Delta U}_{1,opt} - \underline{\boldsymbol{\Delta U}_1} \\ \overline{\boldsymbol{\Delta U}}_1 - \boldsymbol{\Delta U}_{1,opt} \end{bmatrix} \in \mathbb{R}^{2Nc_1 \times 1} \tag{3.72}$$

Considering (3.33) in conjunction with (3.34), it can be seen that the matrix $\tilde{\boldsymbol{E}}\boldsymbol{\psi}$ can be partitioned into

$$\tilde{\boldsymbol{E}} = [\tilde{\boldsymbol{E}}_{U_1}^T \ \tilde{\boldsymbol{E}}_{U_2}^T]^T \tag{3.73}$$

so that,

$$\begin{aligned} \tilde{\boldsymbol{E}}_{U_1} &= [\tilde{\boldsymbol{E}}_{11} \ \tilde{\boldsymbol{E}}_{12}] \\ \tilde{\boldsymbol{E}}_{U_2} &= [\tilde{\boldsymbol{E}}_{21} \ \tilde{\boldsymbol{E}}_{22}] \end{aligned} \tag{3.74}$$

Similarly, $\boldsymbol{K}_R$ can be partitioned into

$$\boldsymbol{K}_R = [\boldsymbol{K}_{R_1} \ \boldsymbol{K}_{R_2}] \tag{3.75}$$

where $\boldsymbol{K}_{R_1} \in \mathbb{R}^{(Np_1+Np_2)\times Np_1}$ is associated with $\boldsymbol{\Gamma}_1$ and $\boldsymbol{K}_{R_2} \in \mathbb{R}^{(Np_1+Np_2)\times Np_2}$ is associated with $\boldsymbol{\Gamma}_2$, respectively. This can be deduced from (3.43).

Now, in the second scenario, $\boldsymbol{\Gamma}_2 \neq \mathbf{0}$ while $\boldsymbol{\Gamma}_1 = \mathbf{0}$, which accounts for a set-point change in y_2 while y_1 is supposed to be constant, hence the optimisation

problem (3.40) becomes

$$\begin{aligned} \underset{\boldsymbol{\Gamma}_2}{min.} \quad & \boldsymbol{\Gamma}_2^T \boldsymbol{\Gamma}_2 \\ s.\,t. \quad & \boldsymbol{M}_{\Delta U_1}^{(2)} \boldsymbol{\Gamma}_2 \leq \boldsymbol{N}_{\Delta U_1} \end{aligned} \tag{3.76}$$

where

$$\boldsymbol{M}_{\Delta U_1}^{(2)} = \begin{bmatrix} \tilde{\boldsymbol{E}}_{U_1} \boldsymbol{K}_{R_2} \\ -\tilde{\boldsymbol{E}}_{U_1} \boldsymbol{K}_{R_2} \end{bmatrix} \in \mathbb{R}^{2Nc_1 \times Np_2} \quad \boldsymbol{N}_{\Delta U_1} = \begin{bmatrix} \boldsymbol{\Delta U}_{1,opt} - \underline{\boldsymbol{\Delta U}_1} \\ \overline{\boldsymbol{\Delta U}}_1 - \boldsymbol{\Delta U}_{1,opt} \end{bmatrix} \in \mathbb{R}^{2Nc_1 \times 1} \tag{3.77}$$

Remark 3.3.2. *In the case of disturbances, which may cause constraint violations, it might be advisable to allow the entire reference trajectory perturbation vector* $\boldsymbol{\Gamma}$ *to take on non-zero values in order to improve disturbance rejection.*

Implementation results are shown in Figure 3.4. It can be observed that cross-coupling effects in the outputs are non existent.

Next, the NMSS MPC using an integral-of-errors state variable, as discussed in Section 3.2.2, is considered. Again, constraints on the increments of the first system input, as above, are imposed.

The constraints are handled by solving the optimisation problem (3.68), accompanied by (3.69). However, since constraints on $\boldsymbol{\Delta U}_1$ are imposed only, the matrices (3.69a) are required to be adapted accordingly, i.e. the constraints associated with $\boldsymbol{\Delta U}_1$ are required to be considered only. The matrices in (3.69a) are adapted to be

$$\bar{\boldsymbol{M}}_{\Delta U_1} = \begin{bmatrix} \bar{\boldsymbol{\Upsilon}}_1 \bar{\bar{\boldsymbol{E}}} \boldsymbol{\psi} \\ -\bar{\boldsymbol{\Upsilon}}_1 \bar{\bar{\boldsymbol{E}}} \boldsymbol{\psi} \end{bmatrix} \quad \bar{\boldsymbol{N}}_{\Delta U_1} = \begin{bmatrix} \underline{\boldsymbol{\Delta U}_1} + \bar{\boldsymbol{\Upsilon}}_1 \bar{\bar{\boldsymbol{E}}} \bar{\bar{\boldsymbol{V}}} \boldsymbol{\psi} - \bar{\boldsymbol{\Upsilon}}_2 \boldsymbol{u}_{k-1} \\ -\overline{\boldsymbol{\Delta U}}_1 - \bar{\boldsymbol{\Upsilon}}_1 \bar{\bar{\boldsymbol{E}}} \bar{\bar{\boldsymbol{V}}} + \bar{\boldsymbol{\Upsilon}}_2 \boldsymbol{u}_{k-1} \end{bmatrix} \tag{3.78}$$

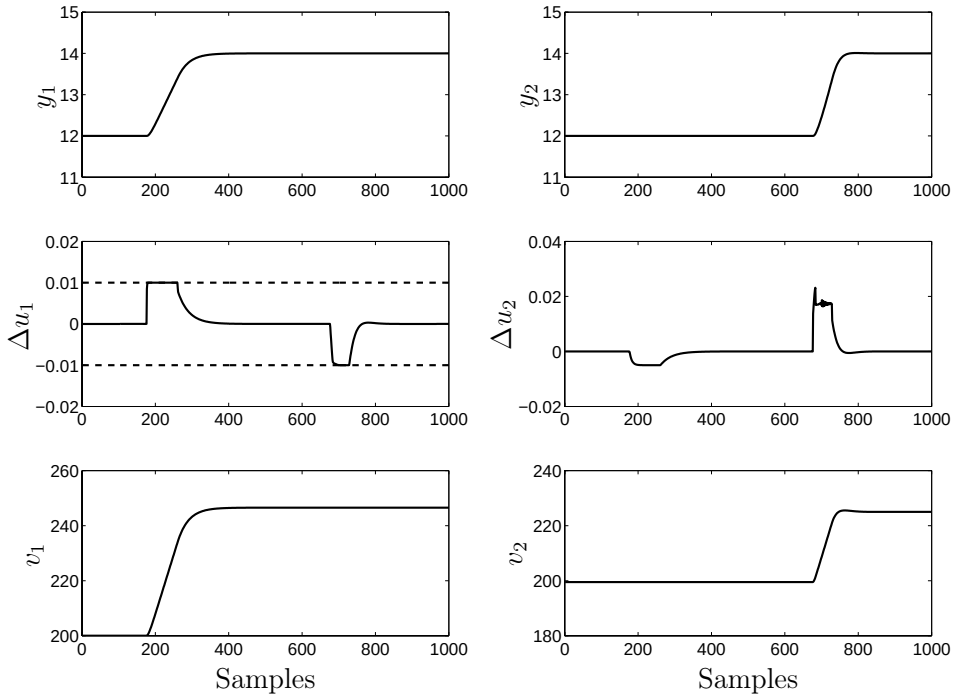


Figure 3.4: Implementation results of the NMSS MPC in incremental input form with imposed constraints

where $\bar{\boldsymbol{M}}_{\Delta U_1} \in \mathbb{R}^{2N_c \times 2N_c}$, $\bar{\boldsymbol{N}}_{\Delta U_1} \in \mathbb{R}^{2N_c \times 1}$ and

$$= \begin{bmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 \end{pmatrix} \\ \vdots \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \end{bmatrix} \in \mathbb{R}^{N_c \times 2N_c} \tag{3.79}$$

such that

$$\boldsymbol{\Delta U}_1 = \quad \boldsymbol{\Delta} \bar{\boldsymbol{U}} \boldsymbol{\psi} \tag{3.80}$$

the incremental input predictions of the first system input are extracted from the entire incremental input prediction vector. The optimisation problem (3.68),

adapted and required to be solved here, is given by

$$\begin{aligned} &\underset{\bar{\boldsymbol{V}}}{min.} \quad J \\ &s.\,t. \quad \bar{\boldsymbol{M}}_{\Delta U_1}\bar{\boldsymbol{V}} \leq \bar{\boldsymbol{N}}_{\Delta U_1} \end{aligned} \tag{3.81}$$

Implementation results are shown in Figure 3.5. It can be observed, and as expected, cross-coupling effects are visible since the constraints involved in the optimisation problem (3.81) do not prevent this, i.e. the optimisation problem (3.81) allows the transformed input v_1 to respond to a set-point change in y_2 and vice versa. This becomes apparent when comparing Figures 3.4 and 3.5. However, it might be possible to construct the constraints in such a way that the cross-coupling effects are eliminated, similarly to (3.71) and (3.76), respectively, without impairing the overall performance significantly. This, however, is left as an open problem.

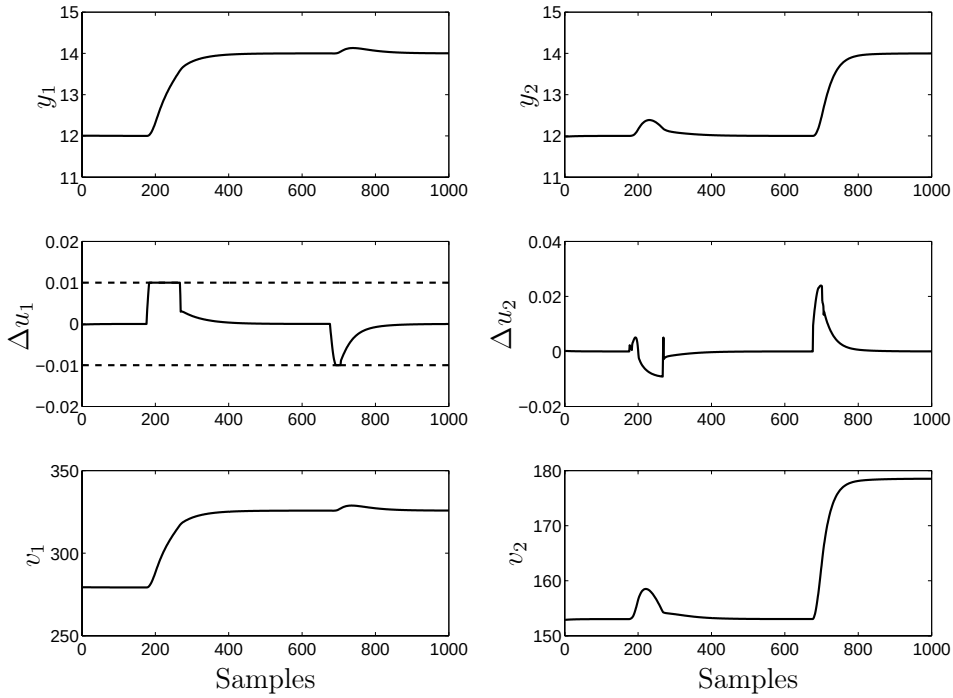


Figure 3.5: Implementation results of the NMSS MPC in integral-of-errors state variable form with imposed constraints

3.4 Sensitivity analysis

In Section 3.3, simulation results were presented where the system model is considered to be mismatch-free. Here, the impact of model mismatch on the performance is evaluated. Other than model mismatch, the simulation set-up is identical to that used in Section 3.3.

Monte-Carlo (MC) simulations are performed, comprising of 1000 runs each. Uniformly distributed random uncertainty terms are added to the nominal system parameters in the range of $\pm 10\,\%$.

The closed-loop poles of the system, in the complex plane, are shown in Figure 3.6. Model parameter uncertainties of the elements in $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$, are considered, respectively. It can be observed that the poles do not lie outside the unit circle. Furthermore, the closed-loop poles of the MPC controllers, in

incremental input form and in integral-of-errors state variable form, respectively, based on the diagonalised model exhibit a wider spread (upper plots) than the closed-loop system poles obtained using the corresponding MPC controllers based on the non-diagonalised model (lower plots). This indicates 'tighter' system output responses of the MC runs. Moreover, the closed-loop poles obtained using the MPC controllers in incremental input form (left-hand plots), based on diagonalised (upper plot) and non-diagonalised (lower plot) system model, are very similar to the respective MPC controllers in integral-of-errors state variable form (right-hand plots). Also, note that uncertainties in $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ only, do not affect the output decoupling properties, i.e. the system diagonalisation, since this is achieved by making use of $\boldsymbol{B}(z^{-1})$ (3.8)–(3.9), as discussed in Section 3.1.2.

Next, consider model parameter uncertainties in the elements of $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, respectively. The corresponding closed-loop poles of the system, in the complex plane, are shown in Figure 3.7. Although barely visible in these plots, some of the poles obtained using the MPC controllers based on the diagonalised model (upper plots), lie outside the unit circle, i.e. resulting in an unstable response. The poles obtained using the MPC controllers based on the non-diagonalised model, however, all lie inside the stable region, i.e. inside the unit circle.

Similar to the observations made concerning the results in Figure 3.6, the closed-loop poles obtained by making use of the MPC controllers based on the diagonalised model (upper plots), are almost identical, as well as the closed-loop poles obtained by making use of the MPC controllers based on the non-diagonalised model (lower plots).

Comparing Figures 3.6 and 3.7, it is observed that the closed-loop poles are generally wider spread in Figure 3.6. However, there are isolated instances of unstable closed-loop poles observable in Figure 3.7. Consequently, it can be said

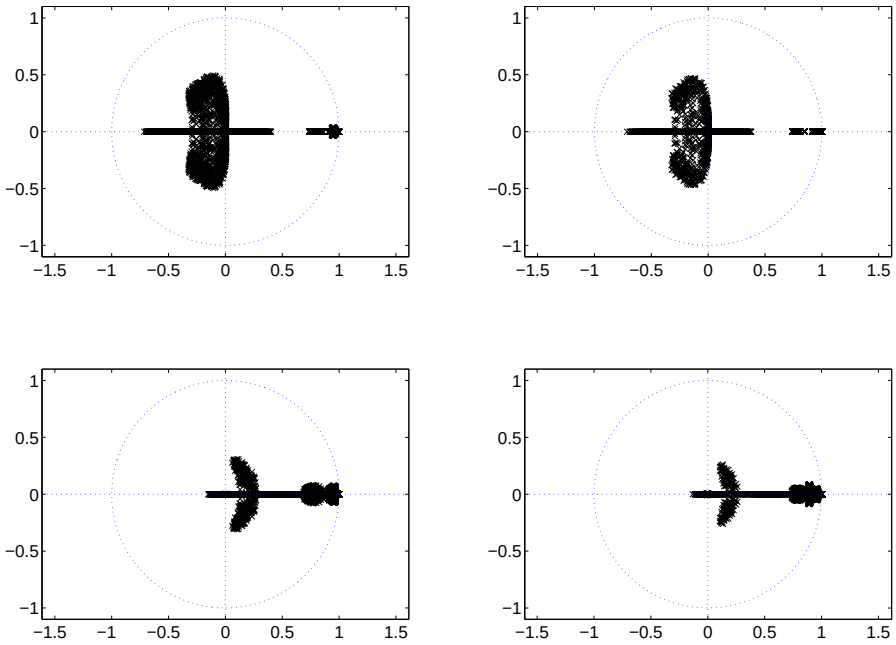


Figure 3.6: Closed-loop poles in the complex plane when model uncertainties in $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are considered, respectively. Upper left: Incremental input form, diagonalised model. Upper right: Integral-of-errors state variable form, diagonalised model. Lower left: Incremental input form, non-diagonalised model. Lower right: Integral-of-errors state variable form, non-diagonalised model.

that:

- Uncertainties in the parameters $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ do have a greater impact on the sensitivity.
- Uncertainties in the parameters $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ affect the stability more if the MPC controllers based on the diagonalised model are used.
- Uncertainties in the parameters $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ impair the system output decoupling.

Remark 3.4.1. *When making use of the integral-of-errors state variable form, these integrators are required to be initialised and, consequently, significant initial*

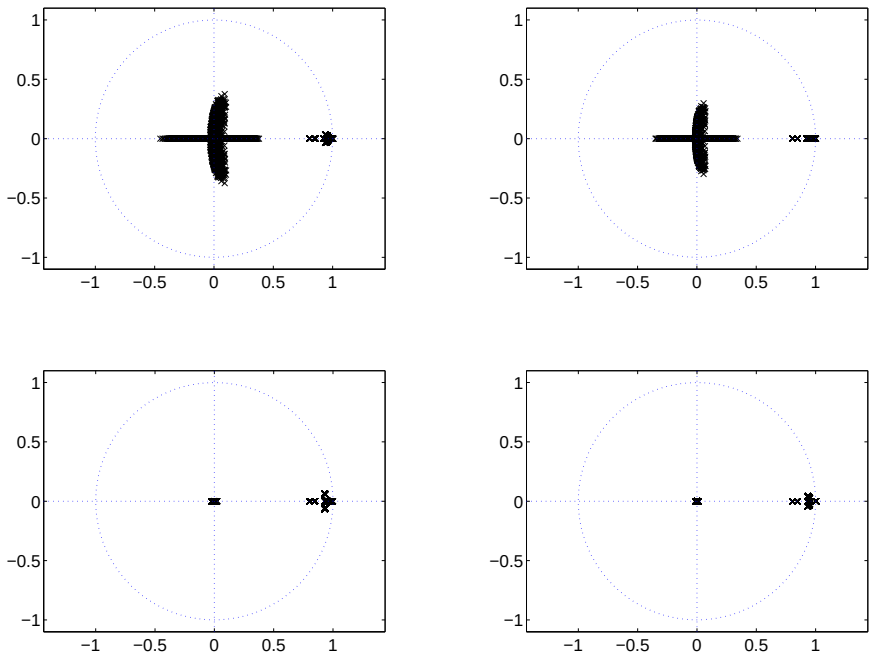


Figure 3.7: Closed-loop poles in the complex plane when model uncertainties in $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ are considered, respectively. Upper left: Incremental input form, diagonalised model. Upper right: Integral-of-errors state variable form, diagonalised model. Lower left: Incremental input form, non-diagonalised model. Lower right: Integral-of-errors state variable form, non-diagonalised model.

system output signal distortion may occur. In this regard, care must be taken in implementing this form of controller.

In order to further illustrate the observations made above, the MC simulation system output responses to model parameter uncertainties in $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$, respectively, in the range of $\pm 5\,\%$ are shown in Figure 3.8, where use is made of the MPC in incremental input form based on the diagonalised model (upper plots) and non-diagonalised model (lower plots). It can be observed that, as mentioned above, the MPC based on the diagonalised model does not exhibit cross-coupling effects, while output cross-coupling effects in the case of the MPC based on the non-diagonalised model are clearly visible.

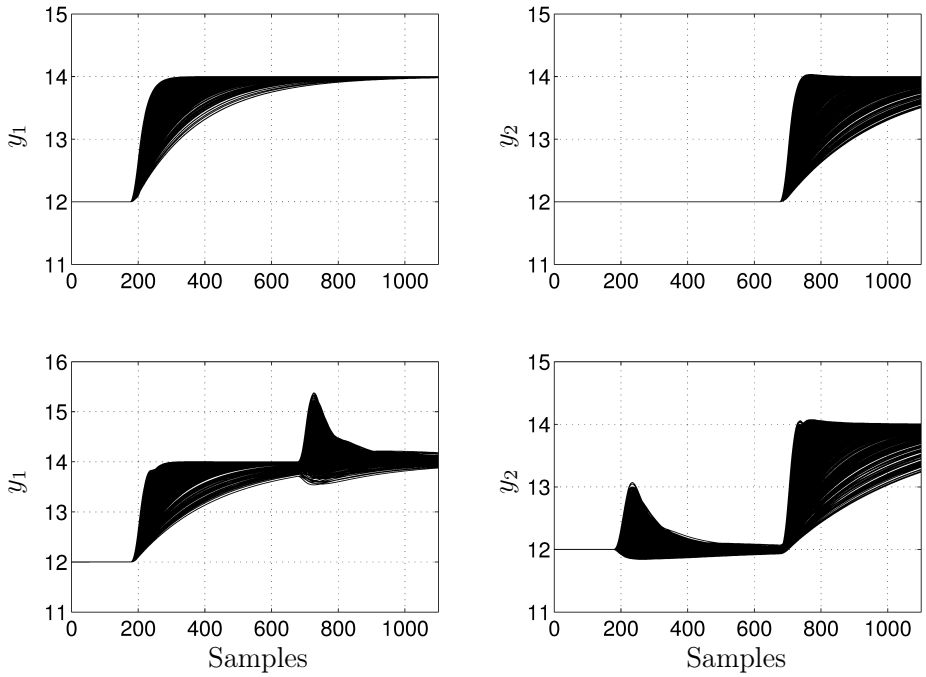


Figure 3.8: MC simulation system output responses when model parameter uncertainties in $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are considered and MPC in incremental input form is applied. Upper plots: MPC based on diagonalised model. Lower plots: MPC based on non-diagonalised model.

MC simulation system output responses to model parameter uncertainties in $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, respectively, in the range of $\pm 5\,\%$ are shown in Figure 3.9, where, again, use is made of the MPC in incremental input form based on the diagonalised model (upper plots) and non-diagonalised model (lower plots). Here, it can be observed that cross-coupling effects re-emerge and that when considering the case of using the non-diagonalised model, compared to Figure 3.8, a 'tighter' performance is achieved, which coincides with the observations made based on the closed-loop pole locations shown in Figures 3.6 and 3.7, respectively.

Remark 3.4.2. *In Figures 3.8 and 3.9, respectively, the MPC in incremental input form is considered only due to the initial output distortion caused by initialising the integral-of-errors state variable. Moreover, model parameter uncer-*

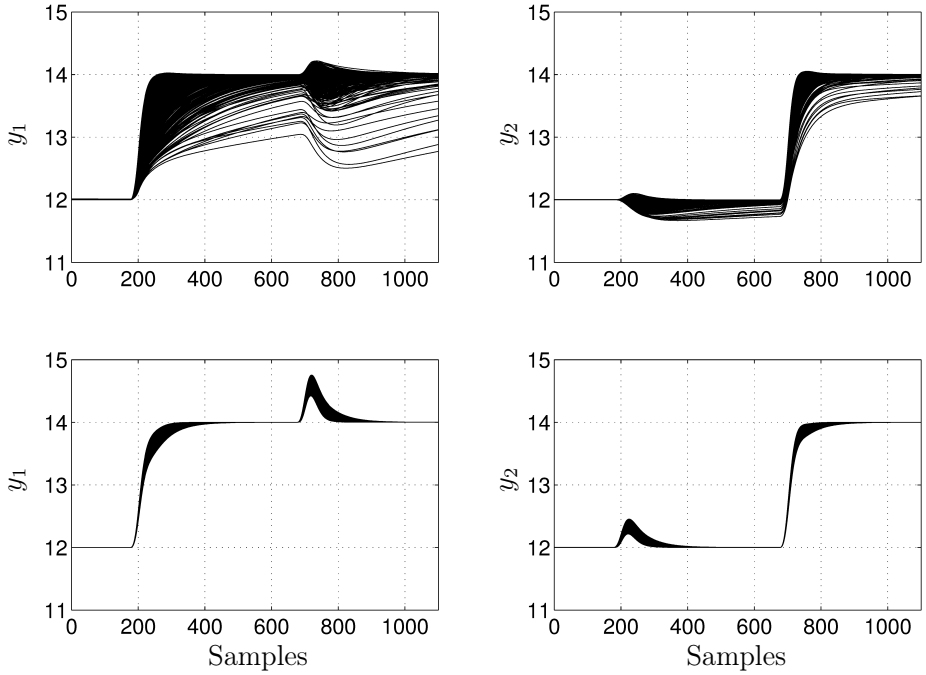


Figure 3.9: MC simulation system output responses when model parameter uncertainties in $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ are considered and MPC in incremental input form is applied. Upper plots: MPC based on diagonalised model. Lower plots: MPC based on non-diagonalised model.

tainties in the range of $\pm 5\,\%$*, instead of* $\pm 10\,\%$*, are considered in order to avoid unstable MC realisations. However, a representative system behaviour is still provided.*

3.5 Concluding remarks

In this chapter, a NMSS MPC approach, eliminating cross-coupling effects in the outputs of square MIMO systems, has been presented. The cross-coupling elimination is achieved by a system diagonalisation method, adopted from (Plummer and Vaughan 1997; Kubalcik and Bobal 2006), where an input transformation is performed, which compensates for the cross-coupling effects. This method is applied to MPC controllers which are based on non-minimal state-space mod-

els using an integral-of-errors state variable (Exadaktylos et al. 2006) and an incremental input form (Wang and Young 2006).

Furthermore, when constraints are imposed, it is desired that the eliminated cross-coupling effects are preserved. For this reason, a modification to the NMSS MPC in incremtal input form has been proposed, which allows this to be achieved in a straightforward manner by enabling the input-output pairs of the MIMO system, i.e. transformed input and output, to be considered individually. Moreover, individual control and prediction horizons can be assigned to each of these pairs, which can be viewed as additional tuning parameters.

Finally, for demonstration purposes, a simulation example has been presented, followed by an evaluation of the impact of model parameter uncertainties.

Chapter 4

Generalised Discrete-time State Dependent Parameter Proportional - Integral - Plus Control

In this Chapter, the state dependent parameter proportional-integral-plus (SDP-PIP) controller in closed-loop is analysed. Particular emphasis is placed on the generalised SDP model structure. Equivalent to its linear counterpart, the SDP-PIP is a pole-placement controller so that it is essential to be able to formulate the closed-loop characteristic equation. It is apparent that linear systems theory (see e.g. Gajic and Lelic 1996; Ogata 1997; Nise 2008), such as the notion and concept of system poles and zeros, cannot necessarily be carried over, nor referred to, when dealing with nonlinear systems and their respective nonlinear system models. However, considering linearised system models at a series of distinct operating points, the notion of equivalent system poles and zeros is often envisaged from an engineering point of view. Such a notion is valid in a restricted region around the considered operating point and it can be said, at least loosely, that the nonlinear system exhibits equivalent poles and zeros, that may be operating point, and/or

time dependent.

Other than control methodologies based on linear models which robustly accommodate the system nonlinearities into the linear model by explicitly including model uncertainties, see e.g. (Kothare et al. 1996; Mayne et al. 2000; Kouvaritakis et al. 2000; Cannon et al. 2003), or by adaptive control approaches, see e.g. (Harris and Billings 1981), or by making use of optimisation based methods, e.g. state dependent Riccati equations (SDRE) (Cloutier 1997; Cimen 2008), the SDP-PIP pole-assignment controller achieves its goal by explicitly cancelling the system nonlinearities and attempts to exactly replicate a desired, linear closed-loop system (Taylor et al. 2009). In (Stables et al. 2006; Stables and Taylor 2006) the time delay is handled by making use of the Smith-predictor (Smith 1959), which, essentially, by predicting the system output based on the system model, 'removes' the time delay from the control loop so that the controller is not affected. While, in (Taylor et al. 2009), the time delay is incorporated into the controller. The major limitation in either approaches lies in the issue that the SDP-PIP is not able to handle numerator zeros of the equivalent linear, or time-step 'frozen', system model. Hence the applicability of the SDP-PIP approach is restricted. However, despite the noted limitations, the SDP-PIP has been successfully implemented in several practical applications, see e.g. (Taylor et al. 2007b, and references therein).

4.1 Aspects on time shift operations

When considering linear PIP and time-invariant parameter system models, the time shift operator is applied to the input/output signals. However, when dealing with the SDP framework, the system model parameters are themselves functions of these time-varying signals, i.e. functions of the non-minimal states. As a

consequence, the time shift operator may be considered to be applied to the parameters as well as the signals.

Throughout this Chapter, for simplicity, the following assumptions are made (unless otherwise stated):

Assumption 1. *There is no mismatch between the system and the model.*

Assumption 2. *The system (and model) is noise free.*

Assumption 3.

- *The system model polynomials* (4.2) *are coprime* $\forall k$
- *The system model* (4.1) *satisfies* $m \leq n_a$*, i.e. proper.*

The system model is obtained as a SDP model in difference equation form

$$y_k = \frac{B_k(z^{-1})\ z^{-\tau}}{A_k(z^{-1})}\ u_k = \frac{B_k(z^{-1})}{A_k(z^{-1})}\ u_{k-\tau} \tag{4.1}$$

where $\tau \in \mathbb{N}$ denotes the sampled time delay and the system model polynomials are given by

$$\begin{aligned} A_k(z^{-1}) &= 1 + a_{1,k}z^{-1} + a_{2,k}z^{-2} + \ldots + a_{n_a,k}z^{-n_a} \\ B_k(z^{-1}) &= b_{\tau,k} + b_{\tau+1,k}z^{-1} + b_{\tau+2,k}z^{-2} + \ldots + b_{\tau+m,k}z^{-m} \end{aligned} \tag{4.2}$$

The subscript k is used to denote the time-varying nature of the polynomials (4.2), indicating the state dependency of the associated parameters. Also, $n_a \in \mathbb{N}_0$ denotes the order of the system model while $m \in \mathbb{N}_0$ denotes the order of the numerator polynomial.

Regarding the SDP system model in the context of the SDP-PIP pole-assignment control approach, it is, in the authors view, important to take certain points into consideration at the outset:

- Since the SDP-PIP is a model-based control methodology, the model is an integral component of the controller. Consequently, the design of the SDP-PIP control approach starts at the system identification stage, i.e. from the definition of the SDP model.

- Since the SDP system model is nonlinear and assigning closed-loop poles can only be performed on linear system models (as already mentioned above), as a consequence, the SDP-PIP controller can only operate on the instantaneous linear (at the current time step 'frozen') system model.

Now, directing attention to the SDP system model (4.1) and, in particular, the role of the time shift operator. When time shifting the entire system (4.1) by $i \in \mathbb{Z}$ sampling instances, the time shift applies to the signals as well as to the system polynomials (cf. Ziemian 2002). This can be summarized as follows:

Proposition 1. *Let k denote the current sampling instance, then, for the subsequent consecutive instances, $y_{k+i} = \dfrac{B_{k+i}(z^{-1})}{A_{k+i}(z^{-1})}\, u_{k+i-\tau}$ $\forall i \in \mathbb{Z}$ holds.*

Demonstration. Let $\hat{k} = k + i,\ i \in \mathbb{Z}$, denote some sampling instance. From (4.1) it follows that if $y_k = \dfrac{B_k(z^{-1})}{A_k(z^{-1})}\, u_{k-\tau}$, then $y_{\hat{k}} = \dfrac{B_{\hat{k}}(z^{-1})}{A_{\hat{k}}(z^{-1})}\, u_{\hat{k}-\tau}$ and substituting $\hat{k} = k + i$ yields $y_{k+i} = \dfrac{B_{k+i}(z^{-1})}{A_{k+i}(z^{-1})}\, u_{k+i-\tau}$. □

In Proposition 1, it is shown that time shifting of the system model affects both the signals and the parameters, indicated by the discrete time subscript on the signals and the system model polynomials, which include the model parameters.

At this juncture it is interesting to explore different interpretations of the time shift operator on the parameters contained in the polynomials (4.2) and

the signals. Re-arranging (4.1) yields the general system representation

$$A_k(z^{-1})\ y_k = B_k(z^{-1})\ u_{k-\tau} \tag{4.3}$$

in which, regarding the time shift operator, two scenarios are of interest.

Scenarios of interest:

1. The parameters as well as the signals are time shifted, i.e. $z^{-1}\ a_{1,k}\ y_k = a_{1,k-1}\ y_{k-1}$, which is similar to the observations and interpretation with the approach proposed in (Ziemian 2002).

2. The signals only are time shifted, i.e. $a_{1,k}\ z^{-1}\ y_k = a_{1,k}\ y_{k-1}$.

Following Scenario 1, and re-arranging (4.3), the system output becomes

$$\begin{aligned} y_k = &-a_{1,k-1}\ y_{k-1} - a_{2,k-2}\ y_{k-2} - \ldots - a_{n_a,k-n_a}\ y_{k-n_a} \\ &+b_{\tau,k}\ u_{k-\tau} + b_{\tau+1,k-1}\ u_{k-\tau-1} + b_{\tau+2,k-2}\ u_{k-\tau-2} \\ &+ \ldots + b_{\tau+m,k-m}\ u_{k-\tau-m} \end{aligned} \tag{4.4}$$

while, following Scenario 2, and re-arranging (4.3), the system output equation becomes

$$\begin{aligned} y_k = &- a_{1,k}\ y_{k-1} - a_{2,k}\ y_{k-2} - \ldots - a_{n_a,k}\ y_{k-n_a} \\ &+ b_{\tau,k}\ u_{k-\tau} + b_{\tau+1,k}\ u_{k-\tau-1} + b_{\tau+2,k}\ u_{k-\tau-2} + \ldots + b_{\tau+m,k}\ u_{k-\tau-m} \end{aligned} \tag{4.5}$$

Both Scenarios are in compliance with Proposition 1 since here, the shift mechanisms are contained inside the polynomials only. When making use of the interpretations according to Scenarios 1 and 2, the system model output quantity y_k obtained from each Scenario, i.e. (4.4) and (4.5), must be identical if both represent models of the same system. Since the state dependent parameters are

functions of the states and these functions can be freely (within the general SDP framework) chosen, effectively, allows usage of either Scenario 1 or Scenario 2 in order to obtain an adequate SDP model of the system.

Consequently, using the time shift operator inside the polynomials according to (4.4) or (4.5) has to be determined at the system identification stage, hence the choice may depend on the system identification method used and, in this regard, which structure is most advantageous.

Remark 4.1.1. *Despite the difference in the notation used in* (4.4) *and* (4.5)*, it should be noted that, in fact, both are mathematically identical when substituting the identified parameters being functions of the states. Since after all, bearing in mind Assumption 1, the quantity* y_k*, obtained from both* (4.4) *and* (4.5)*, must be identical to that of the system.*

When expressing (4.5) in regression vector form, i.e. $y_k = \boldsymbol{\varphi}_k^T \, \boldsymbol{\theta}_k = [-y_{k-1} \ldots - y_{k-n_a} \; u_{k-\tau} \ldots u_{k-\tau-m}] \; [a_{1,k} \ldots a_{n_a,k} b_{\tau,k} \ldots b_{\tau+m,k}]^T$, as it appears in standard system identification methods involving the least-squares algorithm, it can be seen that the parameters required to be identified in the parameter vector $\boldsymbol{\theta}_k$ are related to the same time instance k, where the system output y_k is considered. In the majority of literature, the developed system identification methods targeting SDP models, are formulated such that the time index of the parameters is the instantaneous value of k, for which the system output is considered, hence they make use of the SDP model structure (4.5), cf. Sections 2.2 and 2.3. Consequently, in the remainder of this Thesis, the SDP model structure according to (4.5) is adopted.

Returning to the system polynomials (4.2) and using the system equation in the form of (4.5) in the light of Proposition 1, consideration regarding the interpretation of the time shift operator is now highlighted.

The shift operators 'inside' the polynomials, affect the signals only while shifting the whole system, with the shift operator 'outside' the polynomials, affects both the system model polynomials (hence the parameters) and the signals. Essentially, the different use and interpretation of the time shift operator arises from the way the SDP model and, subsequently, the SDP model structure is defined. In (4.5), the time instance of the current system output serves as a reference time instance to which the parameters are related, so that the time shift operator affects the signals only and not the parameters. Proposition 1 states that when shifting the whole system, the parameters are affected by a time shift, as well.

So, for clarity, a novel, purely conceptual time shift operator for the latter case is proposed. Whereby use of the 'standard' time shift operator z is recognized for time shifting the signals only, the conceptual time shift operator is used for both the signals and the parameters. The conceptual time shift operator is introduced in the following definition.

Definition 4.1.1. *Let $a_{j,k}$ be some arbitrary SDP as it appears in e.g.* (4.3) *and restricting the time shift operator z^i $i \in \mathbb{Z}$ to shift the signals only by i sampling instances, e.g. $a_{j,k}z^2y_k = a_{j,k}y_{k+2}$. While, introducing the conceptual time shift operator $\mathfrak{z}^i$ $i \in \mathbb{Z}$, which shifts the parameters and the signals by i sampling instances, e.g. $\mathfrak{z}^2 a_{j,k}y_k = a_{j,k+2}y_{k+2}$.*

Remark 4.1.2. *Since the time shift operator z according to Definition 4.1.1, arises from the choice of the SDP model structure* (4.5) *and the subsequent way the system identification is performed, different choices of the model structure and system identification methodologies, respectively, may require different interpretations of the shift operator.*

Remark 4.1.3. *In the case of time invariant parameter models, it is not necessary to introduce an additional time shift operator, $\mathfrak{z}^i$, to be applied to the parameters.*

Regarding the two time shift operators, z^i and $\mathfrak{z}^i$, certain observations can be made:

- At every time instance, the SDP system model (4.1) can be seen as an instantaneous linear model which is defined entirely w.r.t the time shift operator z^i affecting the signals only. Hence the SDP-PIP controller is expected to operate on this instantaneous linear model so that pole-assignment, w.r.t. the closed-loop characteristic equation defined in the time shift operator z^i, can be performed.

- In order to proceed in time, from one sampling instance to the next, the time shift operator $\mathfrak{z}^i$ is used, which shifts the entire system model according to Proposition 1, i.e. the parameters as well as the signals are shifted.

- The conceptual time shift operator $\mathfrak{z}^i$ can be interpreted as being associated with the 'overall' nonlinear system model, i.e. transferring the system model between sampling instances, while z^i is associated with the instantaneous linear system model at each sampling instance. As a consequence, $\mathfrak{z}^i$ includes z^i but not vice versa, e.g. $\mathfrak{z}^2 a_{j,k} y_k = a_{j,k+2} z^2 y_k = a_{j,k+2} y_{k+2}$.

Note. *When manipulating the SDP system model* (4.1)*, care must be taken that Proposition 1 is not violated.*

This is best demonstrated by making use of an example.

Example 4.1. Solve (4.1) for the current system input u_k, i.e. the system is required to be shifted forward in time by τ sampling instances. So, one may

attempt to operate on (4.1) with z^τ resulting in, since z affects the signals only, $A_k(z^{-1})y_{k+\tau} = B_k(z^{-1})u_k$. This result clearly contradicts Proposition 1 where it is shown that shifting the system in time affects the signals and the parameters. Moreover, as depicted in the discussion above, the parameters are related to the same time instance as the system model output, i.e. $k+\tau$, which is, here, not the case. Consequently, the system model (4.1) is required to be operated on by $\mathfrak{z}^\tau$ instead of z^τ, leading to $A_{k+\tau}(z^{-1})y_{k+\tau} = B_{k+\tau}(z^{-1})u_k$ which is in accordance with Proposition 1.

4.2 Closed-loop SDP-PIP

Figure 4.1 shows the block diagram of the SDP-PIP configured in closed-loop (Taylor et al. 2009). Here, the system is represented according to (4.1) and (4.2).

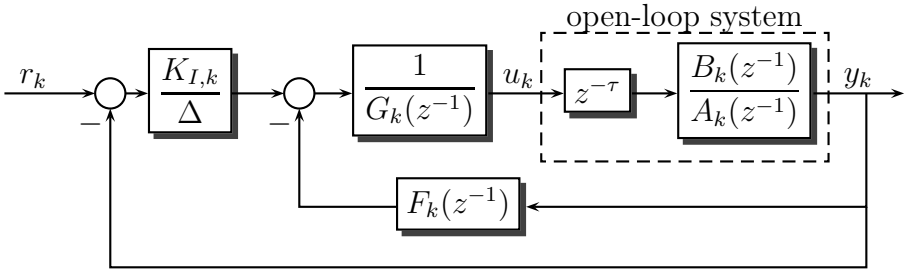


Figure 4.1: Block diagram of the SDP-PIP structure

The controller polynomials are given by

$$\begin{aligned} F_k(z^{-1}) &= f_{0,k} + f_{1,k}z^{-1} + \ldots + f_{n_f,k}z^{-n_f} \quad n_f = n_a - 1 \\ G_k(z^{-1}) &= 1 + g_{1,k}z^{-1} + \ldots + g_{n_g,k}z^{-n_g} \qquad n_g = m + \tau - 1 \end{aligned} \tag{4.6}$$

so that the control law, in polynomial form, is obtained as

$$u_k = -F_k(z^{-1})y_k - \tilde{G}_k(z^{-1})u_k + \frac{K_{I,k}}{\Delta}(r_k - y_k) \tag{4.7}$$

with $\tilde{G}_k(z^{-1}) = G_k(z^{-1}) - 1$ and $K_{I,k} \in \mathbb{R}$ denotes the integral gain. Alternatively, the control law can be formulated in the usual state variable feedback form

$$\begin{aligned} u_k &= -\boldsymbol{c}_k^T \boldsymbol{x}_k \\ \boldsymbol{c}_k &= [f_{0,k}\ f_{1,k}\ \ldots\ f_{n_f,k}\ g_{1,k}\ g_{2,k}\ \ldots\ g_{n_g,k}\ -K_{I,k}]^T \\ \boldsymbol{x}_k &= [y_k\ y_{k-1}\ \ldots\ y_{k-n_a+1}\ u_{k-1}\ u_{k-2}\ \ldots\ u_{k-m-\tau+1}\ \zeta_k]^T \\ \zeta_k &= \frac{r_k - y_k}{\Delta} \end{aligned} \tag{4.8}$$

The integral-of-errors state variable is denoted by ζ_k and the Δ-operator is defined by $\Delta = 1 - z^{-1}$. Furthermore, it is straightforward to see that (4.7) and (4.8) are identical.

For the sake of simplicity, the notation (z^{-1}) indicating function of z^{-1} is dropped from the polynomials (4.2) and (4.6) in the following.

In order to obtain the closed-loop transfer function, the delay is 'removed' from the system by shifting τ sampling instances forward in time, i.e. operating throughout by $\boldsymbol{z}^\tau$, so that the control law (4.7) can be substituted into the open-loop system equation (4.1) in order to 'close the loop', yielding

$$A_{k+\tau} z^\tau y_k + B_{k+\tau} F_k y_k + B_{k+\tau} \tilde{G}_k u_k + B_{k+\tau} K_{I,k} \frac{y_k}{\Delta} = B_{k+\tau} K_{I,k} \frac{r_k}{\Delta} \tag{4.9}$$

To obtain the closed-loop transfer function from r_k to $y_{k+\tau}$, hence the desire to eliminate u_k in (4.9), expanding the term $\tilde{G}_k u_k = \sum_{i=1}^{m+\tau-1} g_{i,k} u_{k-i}$ and substituting (4.1), solved for u_{k-i}, i.e. $u_{k-i} = \dfrac{A_{k+\tau-i}}{B_{k+\tau-i}}\ z^{-i} y_{k+\tau}$ (cf. Example 4.1), the

closed-loop transfer function is then obtained as

$$\frac{y_{k+\tau}}{r_k} = \frac{B_{k+\tau}K_{I,k}}{\Delta\left(\frac{B_{k+\tau}}{B_{k+\tau}}A_{k+\tau} + \sum_{i=1}^{m+\tau-1}\frac{B_{k+\tau}}{B_{k+\tau-i}}g_{i,k}\ A_{k+\tau-i}\ z^{-i} + B_{k+\tau}F_k z^{-\tau}\right) + K_{I,k}B_{k+\tau}z^{-\tau}} \tag{4.10}$$

Certain observations regarding the SDP-PIP closed-loop transfer function can be stated:

- The characteristic equation, i.e. denominator of (4.10), contains transfer functions itself. Hence, it is suspected that the solution of the closed-loop pole-assignment problem does not necessarily yield unique instantaneous controller parameters (if a solution exists).

- The characteristic equation is expressed in terms of the time shift operator affecting the signals only, i.e. z, therefore, it can be anticipated that by assigning desired, stable closed-loop poles (w.r.t z), stability of the instantaneous closed-loop transfer function is achieved by design, cf. (Taylor et al. 2009).

4.2.1 Comparison with linear PIP

It is expected that the SDP-PIP closed-loop transfer function is identical with the closed-loop transfer function obtained for linear PIP since both, linear and SDP-PIP, are of identical structure except that the parameters in the SDP-PIP are time varying while in linear PIP these are time invariant. Consequently, the SDP-PIP comprises the linear PIP as a special sub-class.

It is straightforward to show this relationship. Let the parameters be

time invariant, i.e. $B_{k+i} = B(z^{-1})$ and $A_{k+i} = A(z^{-1})$, it follows that

$$\frac{B_{k+\tau}}{B_{k+\tau}}A_{k+\tau} + \sum_{i=1}^{m+\tau-1}\left(\frac{B_{k+\tau}}{B_{k+\tau-i}}g_{i,k}\ A_{k+\tau-i}\ z^{-i} = A(z^{-1}) + A(z^{-1})\sum_{i=1}^{m+\tau-1}\left(g_i z^{-i}\right.\right.$$
$$= A(z^{-1})\quad 1 + \sum_{i=1}^{m+\tau-1}\left(g_i z^{-i}\right) = A(z^{-1})G(z^{-1})$$

and substituting into (4.10), the linear PIP closed-loop transfer function (2.20) is obtained. Hence linear PIP coexists within the wider SDP-PIP formulation.

4.2.2 Closed-loop SDP-PIP without system zeros

The SDP-PIP in closed-loop without system numerator zeros, i.e. $m = 0$ (or equivalently $B_{k+i} = b_{\tau,k+i}$), as presented in (Taylor et al. 2009), is supposed to be contained within the closed-loop transfer function (4.10) as well (since (4.10) comprises the general case of $m \geq 0$). Substituting $B_{k+i} = b_{\tau,k+i}$ into (4.10) gives

$$\frac{y_{k+\tau}}{r_k} = \frac{b_{\tau,k+\tau}K_{I,k}}{\Delta\left(A_{k+\tau} + \sum_{i=1}^{\tau-1}\left(\frac{b_{\tau,k+\tau}}{b_{\tau,k+\tau-i}}g_{i,k}\ A_{k+\tau-i}\ z^{-i} + b_{\tau,k+\tau}F_k z^{-\tau}\right)\right)\left(+ K_{I,k}b_{\tau,k+\tau}z^{-\tau}\right.} \tag{4.11}$$

From the closed-loop characteristic equation in (4.11), the scaling factor associated with $g_{i,k}$ (4.12), as it appears in (Taylor et al. 2009), becomes apparent.

$$\bar{g}_{i,k} = \frac{b_{\tau,k+\tau}}{b_{\tau,k+\tau-i}}g_{i,k} \tag{4.12}$$

Also, it is straightforward to show that the solution of the pole-assignment problem, obtained by making use of the characteristic equation given in (4.11), is identical to the solution presented in (Taylor et al. 2009).

Consider the desired, instantaneous linear time-invariant characteristic equation

$$D(z^{-1}) = 1 + \sum_{i=1}^{n_a+\tau} d_i \; z^{-i} \tag{4.13}$$

and note that making use of the Δ-operator on the polynomials yields, e.g.

$$\Delta \; A_{k+j} = A_{k+j} - A_{k+j} z^{-1} = \bar{A}_{k+j} \qquad j \in \mathbb{Z} \tag{4.14}$$

where

$$\begin{aligned} \bar{A}_{k+j} &= 1 + \sum_{i=1}^{n_a+1} \bar{a}_{i,k+j} \; z^{-i} \\ \bar{a}_{i,k+j} &= a_{i,k+j} - a_{i-1,k+j} \qquad a_{n_a+1,k+j} = 0 \end{aligned} \tag{4.15}$$

so that finally, the pole-assignment problem is formulated to be

$$\begin{aligned} D(z^{-1}) - \bar{A}_{k+\tau} = &\sum_{i=1}^{\tau-1} \bar{g}_{i,k} \bar{A}_{k+\tau-i} z^{-i} + z^{-\tau}(b_{\tau,k+\tau} - b_{\tau,k+\tau} z^{-1}) F_k \\ &+ K_{I,k} b_{\tau,k+\tau} z^{-\tau} \end{aligned} \tag{4.16}$$

and can be solved by comparing coefficients of like powers of z. In order to solve the pole-assignment problem (4.16) efficiently, it can be formulated in a compact matrix form, see e.g. (Young et al. 1987; Wang and Young 1988; Taylor et al. 2009),

$$\boldsymbol{\beta}_k = \boldsymbol{\Sigma}_k \boldsymbol{\nu}_k \tag{4.17}$$

where $\boldsymbol{\beta}_k \in \mathbb{R}^{(n_a+\tau)\times 1}$ denotes a vector corresponding to the left side of (4.16)

$$\boldsymbol{\beta}_k = [d_1 - \bar{a}_{1,k+\tau} \; d_2 - \bar{a}_{2,k+\tau} \; \ldots \; d_{n_a+1} - \bar{a}_{n_a+1,k+\tau} \; d_{n_a+2} \; \ldots \; d_{n_a+\tau}]^T \tag{4.18}$$

and $\boldsymbol{\Sigma}_k = [\boldsymbol{\Sigma}_{\bar{g},k} \boldsymbol{\Sigma}_{f,k} \boldsymbol{\Sigma}_{K_I,k}] \in \mathbb{R}^{(n_a+\tau)\times(n_a+\tau)}$ denotes a matrix consisting of the three terms of the right side of (4.16) associated with the controller parameters

$g_{i,k}$, $f_{i,k}$ and $K_{I,k}$, respectively. So, $\mathbf{\Sigma}_{\bar{g},k} \in \mathbb{R}^{(n_a+\tau)\times(\tau-1)}$ and $\mathbf{\Sigma}_{f,k} \in \mathbb{R}^{(n_a+\tau)\times n_a}$ yield

$$\mathbf{\Sigma}_{\bar{g},k} = \begin{bmatrix} \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ \bar{a}_{1,k+\tau-1} & 1 & 0 & \dots & 0 \\ \bar{a}_{2,k+\tau-1} & \bar{a}_{1,k+\tau-2} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{a}_{n_a+1,k+\tau-1} & \bar{a}_{n_a,k+\tau-2} & \bar{a}_{n_a-1,k+\tau-3} & \dots & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & \bar{a}_{n_a+1,k+\tau-2} & \bar{a}_{n_a,k+\tau-3} & \dots & 0 \\ 0 & 0 & \bar{a}_{n_a+1,k+\tau-3} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \bar{a}_{n_a,k+1} \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 0 & \dots & \bar{a}_{n_a+1,k+1} \end{pmatrix} \end{bmatrix} \tag{4.19}$$

$$\mathbf{\Sigma}_{f,k} = \begin{bmatrix} \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} b_{\tau,k+\tau} & 0 & \dots & 0 & 0 \\ -b_{\tau,k+\tau} & b_{\tau,k+\tau} & \dots & 0 & 0 \\ 0 & -b_{\tau,k+\tau} & \ddots & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & b_{\tau,k+\tau} & 0 \\ 0 & 0 & \dots & -b_{\tau,k+\tau} & b_{\tau,k+\tau} \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & \dots & 0 & -b_{\tau,k+\tau} \end{pmatrix} \end{bmatrix} \tag{4.20}$$

where the first $\tau - 1$ rows consist of zeros. Finally, with

$$\boldsymbol{\Sigma}_{K_I,k} = [\underbrace{0 \dots 0}_{\tau-1}\ b_{\tau,k+\tau}\ 0 \dots 0]^T \quad \in \mathbb{R}^{(n_a+\tau)\times 1} \tag{4.21}$$

the instantaneous controller parameters contained in the vector

$$\boldsymbol{\nu}_k = [\bar{g}_{1,k}\ \dots\ \bar{g}_{\tau-1,k}\ f_{0,k}\ \dots\ f_{n_a-1,k}\ K_{I,k}]^T \tag{4.22}$$

can be obtained by solving (4.17) for $\boldsymbol{\nu}_k$, i.e.

$$\boldsymbol{\nu}_k = \boldsymbol{\Sigma}_k^{-1}\boldsymbol{\beta}_k \tag{4.23}$$

As expected, the pole-assignment solution (4.23), comprising (4.18)–(4.22), derived from (4.11), is identical to that presented in (Taylor et al. 2009).

4.2.3 SDP-PIP incorporating system zeros

Having shown that, in the case of $m = 0$, the SDP-PIP controller parameters can be derived from the closed-loop transfer function (4.10) and the respective characteristic equation, now, the case of $m > 0$ is explored. Apart from those in (4.11), the scaling factors of $g_{i,k}$ (4.12) are not scalars, instead, it is recognized, these are now transfer functions (cf. (4.10)). In order to deal with this, a similar procedure as that adopted in (4.9) is utilised. Substituting the control law (4.7) into the open-loop system equation (4.1), re-arranged in the form $A_{k+\tau}z^\tau y_k = b_{\tau,k+\tau}u_k + \sum_{i=1}^{m} b_{\tau+i,k+\tau}u_{k-i}$, yields

$$A_{k+\tau}z^\tau y_k = b_{\tau,k+\tau}\left(\left(-F_k y_k - \tilde{G}_k u_k + K_{I,k}\frac{r_k - y_k}{\Delta}\right)\left(+\sum_{i=1}^{m} b_{\tau+i,k+\tau}u_{k-i}\right.\right. \tag{4.24}$$

Again, expanding the term $\tilde{G}_k u_k$ and re-arranging, one obtains

$$\begin{aligned} A_{k+\tau} z^{\tau} y_k + b_{\tau,k+\tau} F_k y_k + b_{\tau,k+\tau} \frac{K_{I,k}}{\Delta} y_k = b_{\tau,k+\tau} \frac{K_{I,k}}{\Delta} r_k \\ + (b_{\tau+1,k+\tau} - b_{\tau,k+\tau} g_{1,k}) u_{k-1} + (b_{\tau+2,k+\tau} - b_{\tau,k+\tau} g_{2,k}) u_{k-2} \\ + \ldots + (b_{\tau+m,k+\tau} - b_{\tau,k+\tau} g_{m,k}) u_{k-m} - g_{m+1,k} b_{\tau,k+\tau} u_{k-m-1} \\ - \ldots - g_{m+\tau-1,k} b_{\tau,k+\tau} u_{k-m-\tau+1} \end{aligned} \tag{4.25}$$

Note, however, that the closed-loop equation (4.25) still contains previous system inputs. In order to eliminate them, from the open-loop system equation (4.1), values of the past system inputs $u_{k-1}, u_{k-2}, \ldots, u_{k-\tau+1}$, are obtained as

$$\begin{aligned} u_{k-1} &= \frac{1}{b_{\tau,k+\tau-1}} A_{k+\tau-1}\ z^{\tau-1} y_k - \frac{b_{\tau+1,k+\tau-1}}{b_{\tau,k+\tau-1}} u_{k-2} - \frac{b_{\tau+2,k+\tau-1}}{b_{\tau,k+\tau-1}} u_{k-3} \\ &\quad - \ \ldots \ - \frac{b_{\tau+m,k+\tau-1}}{b_{\tau,k+\tau-1}} u_{k-m-1} \\ u_{k-2} &= \frac{1}{b_{\tau,k+\tau-2}} A_{k+\tau-2}\ z^{\tau-2} y_k - \frac{b_{\tau+1,k+\tau-2}}{b_{\tau,k+\tau-2}} u_{k-3} - \frac{b_{\tau+2,k+\tau-2}}{b_{\tau,k+\tau-2}} u_{k-4} \\ &\quad - \ \ldots \ - \frac{b_{\tau+m,k+\tau-2}}{b_{\tau,k+\tau-2}} u_{k-m-2} \\ &\vdots \\ u_{k-\tau+1} &= \frac{1}{b_{\tau,k+1}} A_{k+1}\ z\ y_k - \frac{b_{\tau+1,k+1}}{b_{\tau,k+1}} u_{k-\tau} - \frac{b_{\tau+2,k+1}}{b_{\tau,k+1}} u_{k-\tau-1} \\ &\quad - \ \ldots \ - \frac{b_{\tau+m,k+1}}{b_{\tau,k+1}} u_{k-m-\tau+1} \end{aligned} \tag{4.26}$$

and substituting (4.26) into (4.25) gives

$$\begin{aligned} A_{k+\tau} z^{\tau} y_k + b_{\tau,k+\tau} F_k y_k + b_{\tau,k+\tau} \frac{K_{I,k}}{\Delta} y_k = b_{\tau,k+\tau} \frac{K_{I,k}}{\Delta} r_k \\ + \sum_{i=1}^{\tau-1} \gamma_{i,k} A_{k+\tau-i}\ z^{\tau-i} y_k + \sum_{j=1}^{m} \beta_{j,k}\ u_{k-\tau+1-j} \end{aligned} \tag{4.27}$$

where the parameters $\gamma_{i,k} \in \mathbb{R}$ and $\beta_{j,k} \in \mathbb{R}$ arise from sequentially substituting $u_{k-1}, u_{k-2}, \ldots, u_{k-\tau+1}$ from (4.26) into (4.25), hence the computation is iterative

(denoted by the superscripts), details see Appendix A.1.

$$\gamma_{i,k} = \frac{c_i^{(i-1)}}{b_{\tau,k+\tau-i}} \qquad i = 1, 2, \ldots, \tau - 1 \tag{4.28a}$$

$$\beta_{j,k} = c_{\tau+j-1}^{(\tau-1)} \qquad j = 1, 2, \ldots, m \tag{4.28b}$$

$$c_i^{(j)} = \begin{cases} c_i^{(j-1)} - c_j^{(j-1)} \dfrac{b_{\tau+i-j,k+\tau-j}}{b_{\tau,k+\tau-j}} & \text{if } i \leq m + j \\ c_i^{(0)} & \text{else} \end{cases} \tag{4.28c}$$

$$\begin{gathered} c_i^{(0)} = b_{\tau+i,k+\tau} - b_{\tau,k+\tau} g_{i,k} \\ \text{where} \quad b_{\tau+m+l,k+\tau} = 0 \qquad \forall\ l = 1, 2, \ldots \end{gathered} \tag{4.28d}$$

So, the closed-loop characteristic equation can be obtained from (4.27) and by nullifying $\beta_{j,k}$, i.e. forcing $\beta_{j,k}$ such that

$$\beta_{j,k} = 0 \qquad j = 1, 2, \ldots, m \tag{4.29}$$

the term associated with the remaining previous system inputs is eliminated from the closed-loop equation (4.27), so that subsequently, the pole-assignment problem is formulated to be

$$\begin{aligned} D(z^{-1}) - \bar{A}_{k+\tau} = \sum_{i=1}^{\tau-1} (-\gamma_{i,k}) \bar{A}_{k+\tau-i} z^{-i} + z^{-\tau} (b_{\tau,k+\tau} - b_{\tau,k+\tau} z^{-1}) F_k \\ + K_{I,k} b_{\tau,k+\tau} z^{-\tau} \end{aligned} \tag{4.30}$$

which is identical to that obtained for the case of $m = 0$ (4.16), except that here, $\bar{g}_{i,k}$ is replaced by $(-\gamma_{i,k})$. Therefore, (4.23) solves the pole-assignment problem (4.30) as well, if $\boldsymbol{\nu}_k$ is defined to be

$$\boldsymbol{\nu}_k = [-\gamma_{1,k} \ \ldots \ -\gamma_{\tau-1,k} \ f_{0,k} \ \ldots \ f_{n_f,k} \ K_{I,k}]^T \tag{4.31}$$

It is apparent that the instantaneous controller parameters $g_{i,k}, i = 1, 2, \ldots, \tau + m - 1$ cannot be obtained directly. Instead, from (4.28), it can be concluded that $\gamma_{i,k}$ contains the parameters $g_{i,k}, i = 1, \ldots, \tau - 1$ and therefore, $\gamma_{i,k}$ can be seen as the scaled parameters equivalent to (4.12). Effectively, $\gamma_{i,k}$ is a linear function w.r.t the $g_{j,k}, j = 1, 2, \ldots, i$ and due to this linear relationship, it is straightforward to recover the controller parameters $g_{i,k}, i = 1, 2, \ldots, \tau - 1$ from the $\gamma_{i,k}$ by making use of (4.28), see Appendix A.2.

Similarly, the remaining $g_{i,k}, i = \tau, \ldots, \tau + m - 1$ parameters are contained in the $\beta_{j,k}, j = 1, 2, \ldots, m$ which are, in the same manner as $\gamma_{i,k}$, linear functions w.r.t $g_{i,k}, i = \tau, \ldots, \tau + m - 1$. Consequently, it is also straightforward to recover the remaining controller parameters $g_{i,k}, i = \tau, \ldots, \tau + m - 1$ from the $\beta_{j,k}, j = 1, 2, \ldots, m$ by making use of (4.29) in conjunction with (4.28).

Remark 4.2.1. *It is to be noted, that the parameters $g_{i,k}, i = 1, \ldots, \tau - 1$ are obtained from the $\gamma_{i,k}$, while the remaining parameters $g_{i,k}, i = \tau, \ldots, \tau + m - 1$ are obtained from the $\beta_{j,k}, j = i - \tau + 1$.*

Moreover, regarding the role of the $g_{i,k}, i = 1, 2, \ldots, \tau + m - 1$ parameters, the following observations can be made:

- The first $g_{i,k}, i = 1, 2, \ldots, \tau - 1$ parameters of $\tilde{G}_k$ are used for pole-assignment.
    - This follows from (4.31) in conjunction with (4.28) and the discussion above.
- The remaining $g_{i,k}, i = \tau, \tau + 1, \ldots, \tau + m - 1$ parameters of $\tilde{G}_k$ are used for compensating the residuals of the previous time instances.
    - This follows from (4.29) in conjunction with (4.25) and Proposition 1, where it can be seen that the $u_{k-\tau-i}, \forall i = 0, 1, \ldots$ affects past system outputs only.

- The polynomial G_k is used in order to compensate for the influence of the system zeros.

    - Considering the closed-loop equations (4.25) and (4.27) in conjunction with (4.29), it can be seen that the open-loop system polynomial coefficients $b_{\tau+i,k+\tau}, i = 1, \ldots, m$ are eliminated from the closed-loop system numerator, i.e. the system zeros are eliminated.

The resulting closed-loop transfer function, when substituting the controller parameters obtained for the case of $m = 0$ in Section 4.2.2 (also, see Taylor et al. 2009) and for the case of $m > 0$, is identical and of the linear form

$$\frac{y_{k+\tau}}{r_k} = \frac{1 + \sum_{i=1}^{n_a+\tau} d_i}{1 + d_1 z^{-1} + d_2 z^{-2} + \ldots + d_{n_a+\tau} z^{-(n_a+\tau)}} \tag{4.32}$$

In summary, the controller parameters required for the control law (4.8) can be computed in the following manner.

Algorithm 1 SDP-PIP with system numerator zeros

1: Choose desired characteristic closed-loop equation (4.13)
2: Compute $\boldsymbol{\beta}_k$ (4.18) and $\boldsymbol{\Sigma}_k$ (4.19)–(4.21)
3: Solve (4.23) in order to obtain $\boldsymbol{\nu}_k$ (4.31)
4: Obtain $g_{i,k}, i = 1, \ldots, \tau-1$ from $\gamma_{i,k}$ using (4.28) and $g_{i,k}, i = \tau, \ldots, \tau+m-1$ from (4.29) using (4.28)

Example 4.2. Consider the following arbitrarily chosen SDP system where $n_a = 2$, $m = 1$ and $\tau = 2$. Define $a_{1,k} = 0.2e^{y_{k-4}} + 0.1u_{k-3}^2$, $a_{2,k} = -0.3y_{k-3}^3$, $b_{2,k} = -3y_{k-3}$ and $b_{3,k} = 0.5y_{k-4}^{1.5} + 0.3u_{k-4}$. The $n_a + \tau$ desired closed-loop poles are chosen to be located at $p_{1,2} = 0.75 \pm 0.3i$ and $p_3 = p_4 = 0.5$.

According to Algorithm 1, after having chosen the desired closed-loop pole locations and subsequent computation of $\boldsymbol{\beta}_k$ (4.18) and $\boldsymbol{\Sigma}_k$ (4.19)–(4.21), at

every sampling instance k, the parameter values of $\gamma_{1,k}$, $f_{0,k}$, $f_{1,k}$ and $K_{I,k}$ are obtained from (4.31) by solving (4.23). Now, the parameters $g_{i,k}, i = 1, \ldots, \tau - 1$ are calculated from $\gamma_{i,k}$ by making use of (4.28). Note, that here, $i = \tau - 1 = 1$, i.e. $g_{1,k}$ and $\gamma_{1,k}$ are considered. From (4.28a), $\gamma_{1,k} = \dfrac{c_1^{(0)}}{b_{2,k+1}}$ and substituting (4.28d), i.e. $c_1^{(0)} = b_{3,k+2} - b_{2,k+2}g_{1,k}$, yields $g_{1,k} = \dfrac{b_{3,k+2} - \gamma_{1,k}b_{2,k+1}}{b_{2,k+2}}$. Furthermore, the remaining parameters $g_{i,k}, i = \tau, \ldots, \tau + m - 1$ are calculated. Again, note that here, $i = \tau = \tau - m + 1 = 2$. Making use of (4.29) in conjunction with (4.28b) and recall from Remark 4.2.1 that the parameter $g_{i,k}, i = \tau = 2$ is obtained from $\beta_{j,k}, j = i - \tau + 1 = 1$, so that $\beta_{1,k} = 0 = c_2^{(1)}$, which is calculated iteratively. So, using (4.28c), $0 = c_2^{(1)} = c_2^{(0)} - c_1^{(0)}\dfrac{b_{3,k+1}}{b_{2,k+1}}$ and, by taking (4.28d) into account, substituting $c_2^{(0)} = -b_{2,k+2}g_{2,k}$ and $c_1^{(0)} = b_{3,k+2} - b_{2,k+2}g_{1,k}$, re-arranging yields $g_{2,k} = \dfrac{b_{3,k+1}}{b_{2,k+1}}g_{1,k} - \dfrac{b_{3,k+2}b_{3,k+1}}{b_{2,k+2}b_{2,k+1}}$.

The simulated system output compared to the desired closed-loop response (4.32) and the corresponding system input is shown in Figure 4.2. It can be observed that the system output accurately matches the desired response. This is expected since no model mismatch is assumed.

The controller parameters $g_{1,k}$ and $g_{2,k}$ are shown in Figure 4.3, while $f_{0,k}$, $f_{1,k}$ and the integral gain $K_{I,k}$ are depicted in Figure 4.4.

Furthermore, of interest are the roots of G_k in relation to the instantaneous open-loop system zeros. In Figure 4.5, the open-loop system zero (since $m = 1$) and one of the roots of the instantaneous G_k polynomial can be observed, where the response to the first step, yet representative, of the reference signal is shown.

It can be seen in Figure 4.5 (and as indicated above) that the polynomial G_k compensates for the influence of the open-loop system zeros by cancelling them. This may lead to instability in the case of systems exhibiting a non-

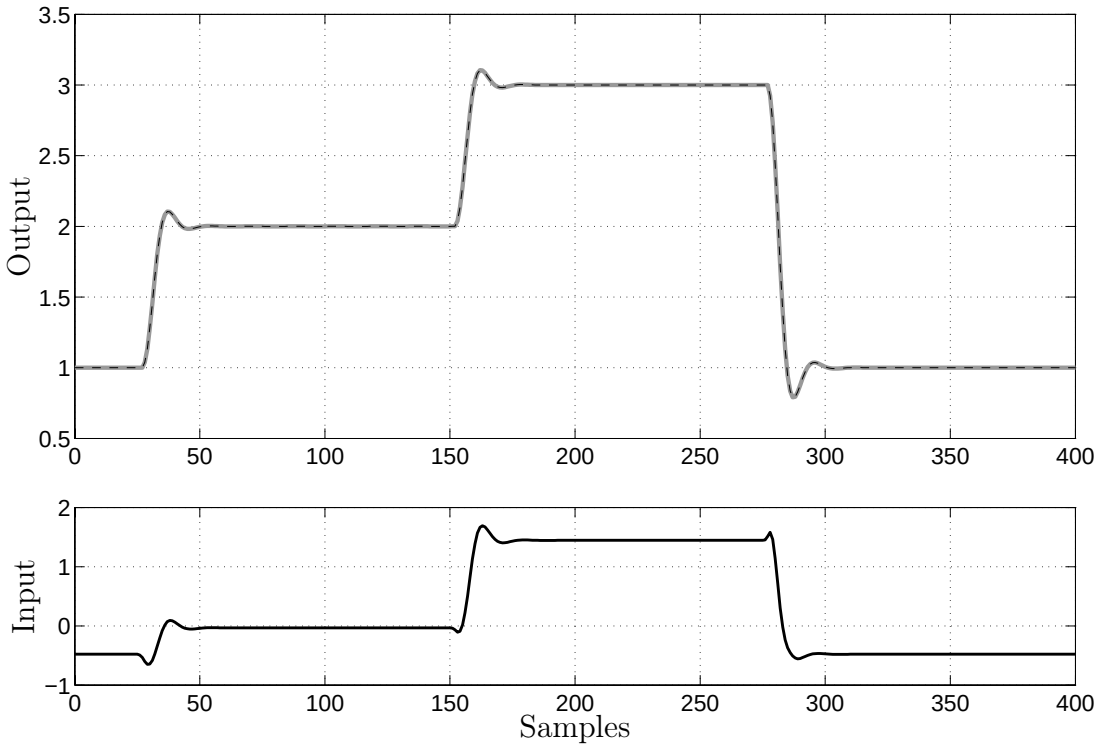


Figure 4.2: Upper: Desired closed-loop system response (black dashed line) and system output (solid grey line). Lower: Corresponding system input.

minimum phase behaviour since the instantaneous system zeros are outside the unit circle. Consequently attempting to cancel these will result in placing poles in the unstable region of the complex plane. This will be especially the case, if there is model mismatch, which is to be expected in a practical, real-world system.

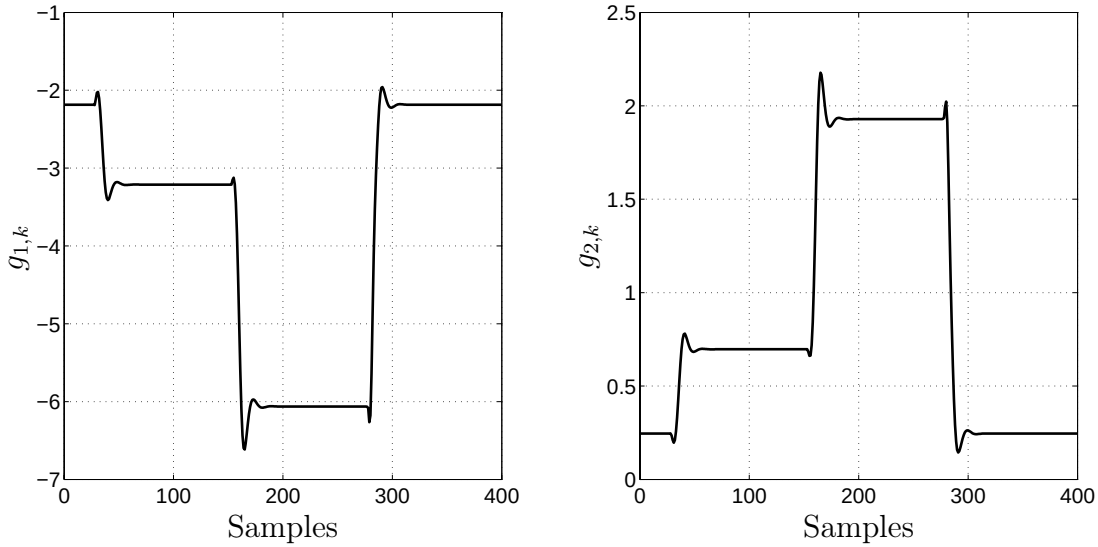


Figure 4.3: Controller parameters $g_{1,k}$ and $g_{2,k}$

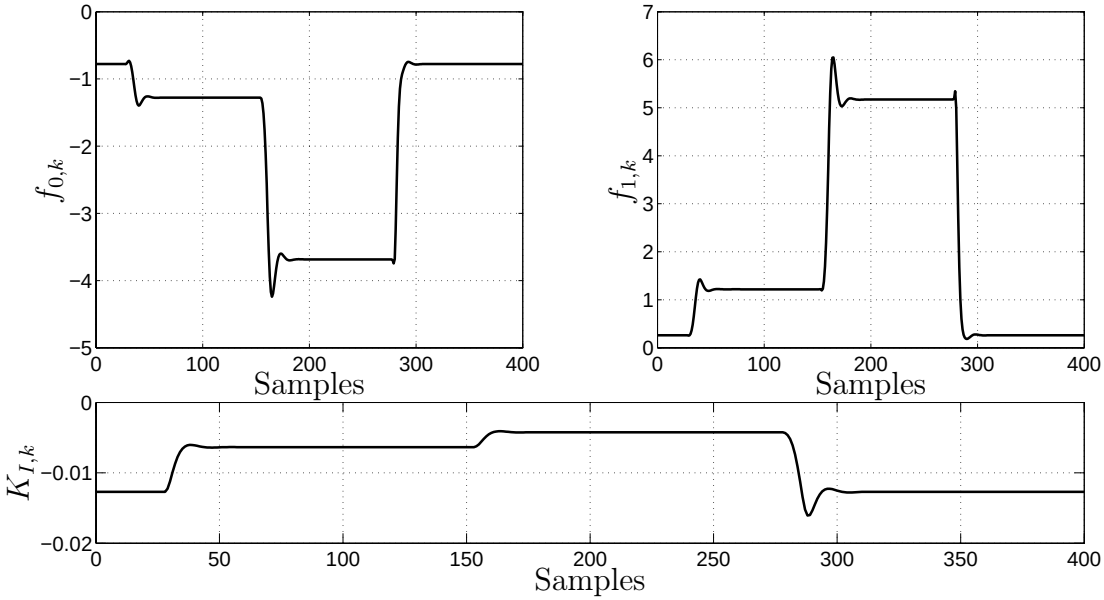


Figure 4.4: The controller parameters $f_{0,k}$, $f_{1,k}$ and $K_{I,k}$.

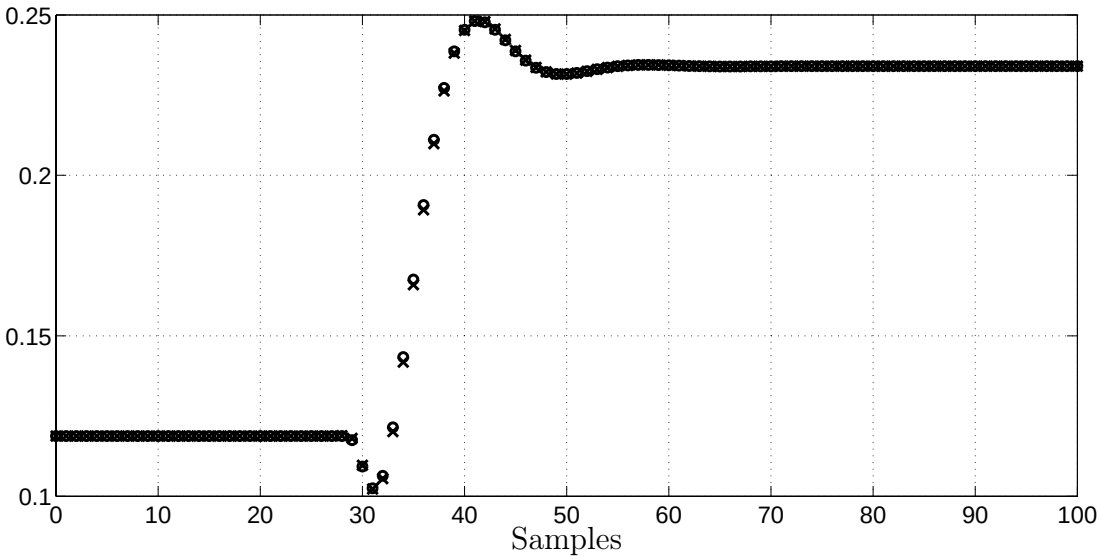


Figure 4.5: Instantaneous open-loop system zeros (circles 'o') and one root of the controller polynomial G_k (cross 'x').

4.3 Sensitivity analysis

The assumptions of no model mismatch and a noise free system presents a rather theoretical scenario and one that could be challenged in practice. Measurement and system noise are always present as well as uncertainties in the model parameters. Especially, since models are by virtue mathematical approximations of real systems, a mismatch free model cannot be obtained in practice. On this basis, this Section considers the influence of noise and model parameter uncertainties on the closed-loop performance.

The evaluation of the robustness to parameter uncertainties and noise is performed by making use of Monte-Carlo (MC) simulation. Computational power is available, nowadays, to perform computationally intensive simulations efficiently; the results presented here are based on 1000 MC simulation runs.

Furthermore, the arbitrarily chosen SDP system in Example 4.2 is used in order to evaluate the effect of parameter uncertainties and noise, whereby a

unit step change in the reference signal, from 1 to 2, is considered (see first step in Figure 4.2).

Other than linear system models, SDP models can be more complex and diverse in their appearance, hence general statements cannot necessarily be made so that every system is required to be considered individually. However, since the SDP-PIP controller is of a certain structure, there can be structurally imposed observations made which are most likely to affect the closed-loop performance.

The performance criteria of choice are the mean squared errors

$$MSE = \frac{1}{N}\sum_{k=1}^{N}(y_k - y_{d,k})^2 \tag{4.33}$$

and the integral of absolute errors

$$IAE = \frac{1}{N}\sum_{k=1}^{N}|y_k - y_{d,k}| \tag{4.34}$$

where $y_{d,k}$ denotes the desired closed-loop system output.

4.3.1 Model parameter uncertainties

Firstly, uncertainties on the system parameters $a_{1,k+i}$ and $a_{2,k+i}$ $i = 1, 2$, that are required for calculating u_k, are considered. The uncertainties of these model parameters are simulated to be uniformly distributed in the ranges of $\pm 5\,\%$, $\pm 1\,\%$ and $\pm 0.36\,\%$ of the nominal parameter values. The results obtained are shown in Table 4.1. It can be observed that almost $90\,\%$ of the MC runs result in an unstable response when the uncertainty was $\pm 5\,\%$. By trial and error, it is found that stable results of all MC runs are obtained when the parameter uncertainties remain in the range of $\pm 0.36\,\%$. This means, that even slight uncertainties, e.g. $\pm 1\,\%$, can lead to instability of the closed-loop system.

uncertainty range [%]	$IAE \times 10^3$ of stable MC runs		$MSE \times 10^3$ of stable MC runs		unstable MC runs [%]
	mean	max	mean	max	
± 5	9.46	136.8	1.68	28.87	88.2
± 1	8.28	225.42	1.6	91.53	39.6
± 0.36	1.69	21.83	0.11	1.36	0

Table 4.1: Monte-Carlo simulation evaluation of model parameter uncertainties in $a_{1,k+i}$ and $a_{2,k+i}$ $i = 1, 2$.

Now, uncertainties in the parameters $b_{2,k+i}$ and $b_{3,k+i}$ $i = 1, 2$ are considered only. Table 4.2 shows results of the MC simulation. Again, uniformly distributed uncertainties in the ranges of $\pm 5\,\%$, $\pm 1\,\%$ and $\pm 0.71\,\%$ of the nominal parameter values are considered. Compared to the results obtained in Table 4.1, more stable MC runs are achieved. Moreover, at an uncertainty range of $\pm 0.71\,\%$ all MC runs were found to result in a stable response. This is almost twice the uncertainty range as achieved in Table 4.1. However, it is still a very small range and it may be questionable whether this would be achievable in a practical application.

Remark 4.3.1. *It is to be noted that the above observations apply only to the specific simulation example considered here and therefore are not general (as mentioned earlier). Moreover, factors such as the chosen, desired closed-loop pole locations, set-point sequences, the nature of the SDPs, etc., all contribute to the robustness/sensitivity properties and by a careful choice of some of these factors, further improvements might be achievable.*

Nevertheless, for this particular system, uncertainties in A_k seem to have a greater impact on stability than uncertainties in B_k. This is also reflected in the performance criteria IAE and MSE where a wider spread of these values can be observed in Table 4.2 than in Table 4.1, despite there being less unstable MC runs obtained in Table 4.2. This result is not surprising as A_k is directly

involved with the dynamical response of the system.

uncertainty range [%]	$IAE \times 10^3$ of stable MC runs		$MSE \times 10^3$ of stable MC runs		unstable MC runs [%]
	mean	max	mean	max	
± 5	25.44	294.53	7.43	140.71	70.9
± 1	2.62	94.84	0.17	12.73	7.9
± 0.71	1.71	49.3	0.091	4.61	0

Table 4.2: Monte-Carlo simulation evaluation of model parameter uncertainties in $b_{2,k+i}$ and $b_{3,k+i}$ $i = 1, 2$.

4.3.2 Measurement noise

In a practical application, measurement noise is unavoidable, so, it is essential to evaluate its influence. Here, normally distributed white measurement noise is assumed. Since in a SDP system model, the parameters are state dependent, which includes the system output, it can be seen as introducing additional, time varying model parameter uncertainties. The model parameter uncertainties, considered in Section 4.3.1 were constant since the parameters (or the coefficients within the parameters) are identified beforehand and do not change subsequently (during runtime). However, here, since the parameters are re-calculated at every time instance based upon measurements that are corrupted by noise, the uncertainties are time varying.

Table 4.3 shows the results of MC simulations, corresponding to different noise variances denoted σ^2. Stable responses of all MC runs are found to be at a variance as low as $\sigma^2 = 16 \times 10^{-6}$. This low noise variance also questions whether this would be achievable in practice. Certainly, a SDP system model where the parameters are less dependent on measured signals may achieve better performance, however, noise remains an issue of concern within a SDP system model.

Remark 4.3.2. *Again, in a similar manner as Remark 4.3.1, it is to be noted that the specificity of the example system considered here, does not necessarily allow the observations to be regarded as general.*

measurement noise variance σ^2	$IAE \times 10^3$ of stable MC runs		$MSE \times 10^3$ of stable MC runs		unstable MC runs [%]
	mean	max	mean	max	
1×10^{-3}	–	–	–	–	100
50×10^{-6}	266.42	271.15	133.85	138.72	99.7
35×10^{-6}	228.93	280.31	99.52	159.11	67.2
25×10^{-6}	183.33	231.44	63.08	209.17	12.1
16×10^{-6}	136.56	168.1	33.57	57.55	0

Table 4.3: Monte-Carlo simulation results obtained for different measurement noise variances.

Furthermore, regarding the performance criteria IAE and MSE in Tables 4.1, 4.2 and 4.3, it can be observed that in the case of measurement noise (Table 4.3) these are the greatest. Recalling that noise can be viewed as a source of time varying uncertainties, the integrator of the SDP-PIP controller does not have sufficient time to drive the output to its desired value since the disturbing uncertainties, i.e. the noise, changes at every sampling instance. This differs from the case of constant uncertainties, where the integrator is able to 'compensate' for these in steady state. In Figure 4.6, the MC simulations are shown where it can be observed that in the case of constant uncertainties of the parameters, the output response decays to the steady-state value, while, in the case of measurement noise, the system output response is steadily excited.

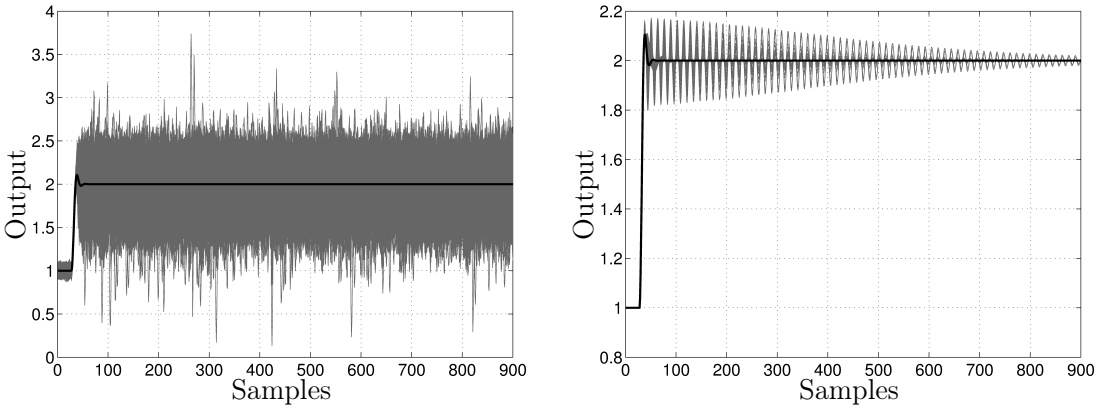


Figure 4.6: Monte-Carlo simulations with measurement noise variance $\sigma^2 = 16 \times 10^{-6}$ (left) and uncertainties on $b_{2,k+i}$ and $b_{3,k+i}$ in the range of $\pm 0.71\,\%$ (right).

4.3.3 Comparison with linear PIP

The analysis presented above is based on an arbitrarily chosen SDP modelled system introduced in Example 4.2, which exhibits severe nonlinearities such that in the presence of model mismatch, the proposed SDP-PIP control approach experiences difficulties and linear PIP control is not able to achieve stable performance (to the Authors experience), hence comparison with linear PIP is not possible when making use of this system. For this reason and in order to account for the fact that nonlinearities to this severe extent are not always present in a practical system, consider the following system taken from (Shaban and Taylor 2006) and slightly adapted as in the following example:

Example 4.3. Consider the following SDP system where $n_a = 2$, $m = 1$ and $\tau = 1$. The model parameters are given by $a_1 = -0.9$, $a_2 = 0.08$, $b_1 = 0.5$ and $b_{2,k} = -0.4u_{k-2}$. As in Example 4.2, the $n_a + \tau$ desired closed-loop poles are chosen to be located at $p_1 = p_2 = 0.75 \pm 0.3i$ and $p_3 = 0.5$. Note that this SDP system model consists of almost only constant parameters, apart from $b_{2,k}$, so that the state variable feedback gains $f_{0,k}$, $f_{1,k}$ and $K_{I,k}$ of the SDP-PIP, calculated

by making use of (4.23) in conjunction with (4.20) and (4.21), are also, in fact, constant. The SDP-PIP controller parameter $g_{1,k}$, however, obtained from (4.28) in conjunction with (4.29), yields $g_{1,k} = \frac{b_{2,k+1}}{b_1}$.

The constant SDP-PIP controller parameters are obtained as $f_{0,k} = -0.3525$, $f_{1,k} = 0.4925$ and $K_{I,k} = 0.1525$. In order to obtain the linear PIP controller parameters, an operating point is required upon which they are based. Since the reference set-point is chosen to be a step from zero of magnitude 0.3, the corresponding steady-state input is obtained to be $u = 0.1194$ so that for the linear PIP controller parameter calculation, $b_2 = -0.4 \cdot 0.1194$ is used. Consequently, the linear PIP controller parameters yield $f_0 = -14.2087$, $f_1 = 11.5914$, $g_1 = 6.9201$ and $K_I = 0.1686$.

Simulation results of the linear PIP (thin solid line) and SDP-PIP (dashed line) are shown in Figure 4.7, where a mismatch free system model is considered. It is observed that the SDP-PIP exactly tracks the desired output response (thick grey solid line), as expected, while the linear PIP produces an oscillatory response which eventually settles to achieve a stable response.

Next, MC simulation is performed and model parameter uncertainties in both numerator and denominator, i.e. in b_1 and $b_{2,k}$ as well as a_1 and a_2, parameters are considered. MC simulation results obtained by making use of the SDP-PIP and model parameter uncertainties in the range of $\pm 19\,\%$ of the nominal parameter values is shown in Figure 4.8. When linear PIP is considered, unstable MC simulation realisations occur at a model parameter uncertainty range of as low as $\pm 1.6\,\%$. The SDP-PIP, on the contrary, is able to handle parameter uncertainties in a range of up to $\pm 19\,\%$. These values are found by trial and error.

This range of parameter uncertainty is likely to be achieved in a practical

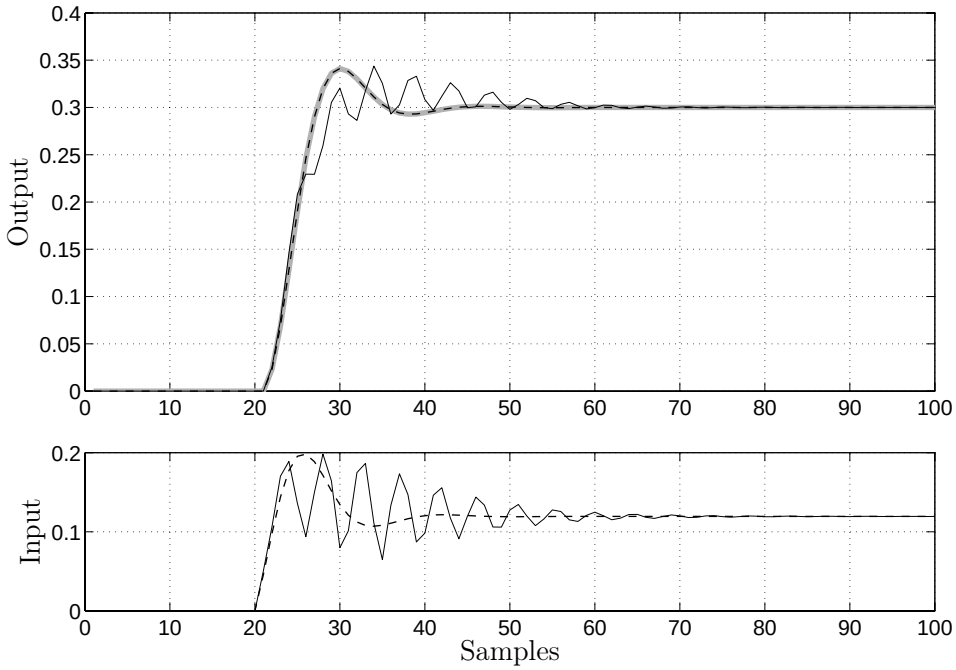


Figure 4.7: Upper: model mismatch free simulation results of the system output of SDP-PIP (dashed line), linear PIP (thin solid line) and the desired response (thick solid line). Lower: corresponding control actions, i.e. system inputs.

application so that the proposed SDP-PIP can be used in a real-world application if the impact of the nonlinearities are moderate. However, it is demonstrated that the linear PIP is clearly outperformed.

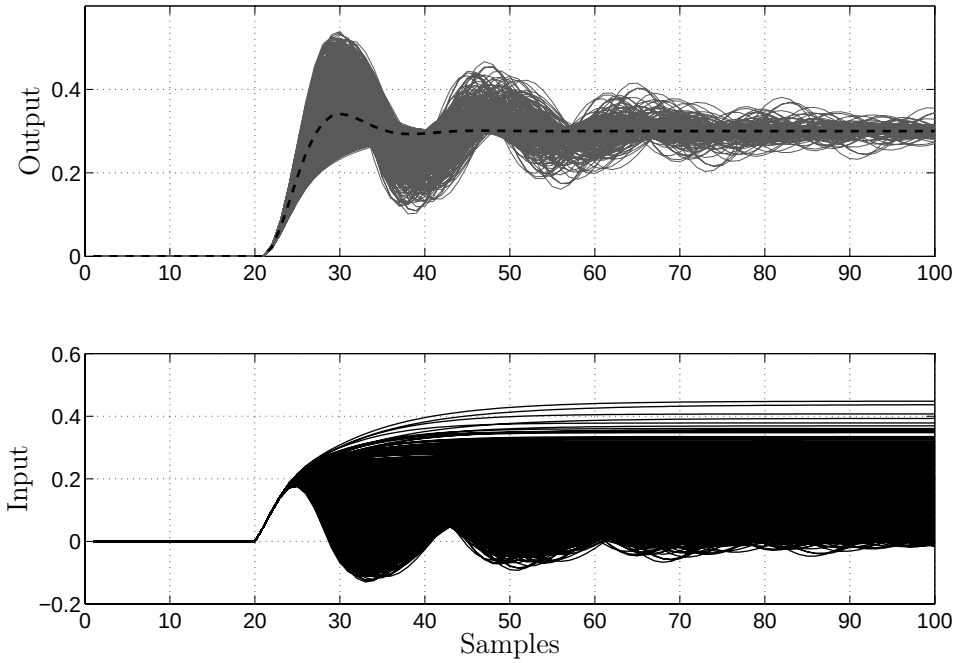


Figure 4.8: MC simulation results for SDP-PIP when model parameter uncertainties in the range of $\pm 19\,\%$ is considered.

4.4 The SDP-PIP in incremental input form

The usage of an integral-of-errors state inflicts certain practical implementation issues such as integrator wind-up, overflow, etc. In order to overcome these issues, (Taylor et al. 2009) proposes to express the control law in an incremental input form so that the necessity of implementing an integral-of-errors state is eliminated. The calculation of the controller parameters, however, is not altered.

Recall from (4.7) and (4.8), respectively, that the control law is given by

$$u_k = -\tilde{G}_k \; u_k - F_k \; y_k + K_{I,k} \; \zeta_k \tag{4.35}$$

where ζ_k denotes the integral-of-errors state. By taking the difference of the current and previous input, the incremental input form is obtained. But note that the input of the previous sampling time instance is required, hence operating

on (4.35) with $\mathfrak{z}^{-1}$, as discussed in Section 4.1, yields

$$u_{k-1} = -\tilde{G}_{k-1}\ u_{k-1} - F_{k-1}\ y_{k-1} + K_{I,k-1}\ \zeta_{k-1} \tag{4.36}$$

and subtracting (4.36) from (4.35), gives

$$u_k = u_{k-1} - \tilde{G}_k u_k + \tilde{G}_{k-1} u_{k-1} - F_k y_k + F_{k-1} y_{k-1} + K_{I,k}\zeta_k - K_{I,k-1}\zeta_{k-1} \tag{4.37}$$

or, equivalently,

$$u_k = u_{k-1} - \tilde{G}_k u_k + \tilde{G}_{k-1} u_{k-1} - F_k y_k + F_{k-1} y_{k-1} + \underbrace{(1-\mathfrak{z}^{-1})K_{I,k}\zeta_k}_{=\hat{\Delta}K_{I,k}\zeta_k} \tag{4.38}$$

where it can be observed that both time varying quantities, the integral-of-errors state and the associated integral gain, are affected by the difference operator, defined to be $\hat{\Delta} = 1 - \mathfrak{z}^{-1}$. Moreover, note that the difference operator $\hat{\Delta}$ differs from Δ that is defined in ζ_k. While $\hat{\Delta}$ takes the difference of the whole system, i.e. signals and parameters, between consecutive sampling instances, Δ operates on the instantaneous linear system only, i.e. on the signals.

Remark 4.4.1. *Similarly as shown in Example 4.1, operating on* (4.35) *with* Δ *is misleading. Since then, the incremental input of the instantaneous linear system is obtained, but, if the instantaneous linear system of the previous sampling time instance differs from the current one, the incremental input of the overall system is, as a consequence, different as well.*

So, in order to eliminate ζ_k, the integral gain $K_{I,k}$ is required to be time invariant. Then, the term $K_{I,k}\zeta_k - K_{I,k-1}\zeta_{k-1}$ in (4.37) becomes $K_I\zeta_k - K_I\zeta_{k-1} = K_I(1-z^{-1})\zeta_k = K_I\Delta\zeta_k = K_I(r_k - y_k)$ since, from (4.8), $\zeta_k = \frac{r_k - y_k}{\Delta}$. Consequently, time invariant integral gains in (4.35) and (4.36) are required in order to obtain

time invariant integral gains in (4.37).

From the solution of the pole assignment problem (4.23), as well as by comparing (4.25) with (4.32), the integral gain $K_{I,k}$ is obtained by

$$K_{I,k} = \frac{1 + \sum_{i=1}^{n_d} d_i}{b_{\tau,k+\tau}} \tag{4.39}$$

where it can be seen that the time varying nature is introduced by the denominator term, i.e. $b_{\tau,k+\tau}$. Hence, first multiplying (4.35) with $b_{\tau,k+\tau}$ and (4.36) with $b_{\tau,k+\tau-1}$ and then subtracting, yields

$$\begin{aligned} b_{\tau,k+\tau}u_k = {} & b_{\tau,k+\tau-1}u_{k-1} - b_{\tau,k+\tau}\tilde{G}_k u_k + b_{\tau,k+\tau-1}\tilde{G}_{k-1}u_{k-1} \\ & - b_{\tau,k+\tau}F_k y_k + b_{\tau,k+\tau-1}F_{k-1}y_{k-1} + \underbrace{\bar{K}_I\zeta_k - \bar{K}_I\zeta_{k-1}}_{=\bar{K}_I\Delta\zeta_k=\bar{K}_I(r_k-y_k)} \end{aligned} \tag{4.40}$$

where $\bar{K}_I = 1+\sum_{i=1}^{n_d} d_i$ denotes the time invariant integral gain. Finally, dividing (4.40) by $b_{\tau,k+\tau}$ and taking (4.39) into account, the control law in incremental input form is obtained by

$$\begin{aligned} u_k = {} & \frac{b_{\tau,k+\tau-1}}{b_{\tau,k+\tau}}\ u_{k-1} - \tilde{G}_k\ u_k + \frac{b_{\tau,k+\tau-1}}{b_{\tau,k+\tau}}\ \tilde{G}_{k-1}\ u_{k-1} \\ & - F_k\ y_k + \frac{b_{\tau,k+\tau-1}}{b_{\tau,k+\tau}}\ F_{k-1}\ y_{k-1} + K_{I,k}\ (r_k - y_k) \end{aligned} \tag{4.41}$$

where the necessity of implementing the integral-of-errors state ζ_k in a practical application is eliminated. Only the controller parameter values of the previous time instance are required to be stored, along with the previous system outputs and inputs.

Identical numerical simulation results are obtained when making use of (4.41) applied to Examples 4.2 and 4.3. This is expected since the controller parameters are also calculated according to Algorithm 1. Hence they are identical

in both cases. Basic algebraic manipulations of the control law is performed after obtaining the controller parameters so that, nevertheless, identical results are obtained.

Finally, note that the control law in incremental input form (4.41) derived here, coincides with that obtained in (Taylor et al. 2009).

4.5 Concluding remarks

This Chapter has been concerned with discrete-time state-dependent parameter proportional-integral-plus pole-assignment control. A conceptual approach has been proposed, which introduced an additional conceptual time shift operator to the commonly used one in order to allow separate consideration of the instantaneous linear model, where pole-assignment has been applied to at each sampling instance, and the overall nonlinear SDP model. It has been shown that when using this underlying concept, further clarification of the SDP-PIP pole-assignment method has been achieved.

Furthermore, an extension to the SDP-PIP pole-assignment approach has been proposed, which targets SDP system models that incorporate system zeros. This approach is based on a cancellation method so that the effects of the system zeros have been eliminated. However, this approach is limited to systems of a minimum phase behaviour since the system zeros appearing in the instantaneous linear model at each sampling instance are cancelled.

It has also been shown that this approach is sensitive to both system model parameter uncertainties and the effect of measurement noise. The model parameter uncertainty sensitivity is dependent on the extent of the system nonlinearities, but also on factors such as the chosen desired closed-loop pole locations, set-point sequence and the nature of the SDPs. However, these factors have not

been investigated in this Chapter and, therefore, remain as a potential direction for further research. In the case of sufficiently moderate nonlinearities, however, it has been demonstrated that the proposed SDP-PIP controller clearly outperforms a linear PIP controller.

Chapter 5

SDP Modelling of a DC-DC Boost Converter Operating in DCM

This Chapter is concerned with modelling a practical, laboratory based application example, namely, a DC-DC boost converter, operating in discontinuous conduction mode (DCM). DC-DC boost converters are switched-mode power electronic devices, that step-up a DC input voltage to a higher DC output voltage. The challenge in terms of modelling a DC-DC switched-mode converter arises from its hybrid nature due to the switching process. Consequently, two conditions are required to be considered, namely: when the switch is open and when the switch is closed. In DCM operation, however, an additional condition is introduced, namely, when the switch is open and the inductor is not conducting. Details on the operational principles are given in Section 5.1.1.

The modelling approach proposed here, is to make use of the state-dependent parameter (SDP) framework in order to obtain a model of the converter. In the proposed approach, the SDP model is based on measured input-output data only, rather than on physical relationships, e.g. circuit components.

Modelling of a system in the context considered here is for the purpose

of model-based control, and the model developed here is subsequently used in order to build the foundation of a model-based SDP-PIP control strategy. The purpose of the strategy is to control the output voltage under various loading scenarios, i.e. varying output current drawn from the converter.

Furthermore, due to the close relationship of a bilinear model with the SDP model structure (Taylor et al. 2011), in Section 5.3, the proposed SDP model is compared with a modelling approach based on a Hammerstein-Bilinear structure (HBS) (Larkowski and Burnham 2011; Larkowski et al. 2013), which is also based on measured input-output data.

As an alternative to modelling DC-DC converters based on measured input-output signals, the pioneering work of (Middlebrook and Cuk 1976) proposed the method of state-space averaging, which is based on physical relationships. For each switch condition, a state-space model is derived separately and, subsequently, these are averaged over the entire switching period. In DCM operation, however, it is not a trivial task to determine the duration of the above mentioned additional condition where the inductor is not conducting. But, this condition is required for averaging. Further information regarding the modelling of a DC-DC boost converter operating in DCM, via this approach, can be found in, e.g. (Maksimovic and Cuk 1991; Sanders et al. 1991; Sun et al. 2001; Davoudi et al. 2006).

An exact physical insight into the converter, however, is not necessarily available due to the tolerances and inherent parasitic elements of the components used. Some values of the parasitics are given in the respective datasheets, but, often, these are of a nonlinear nature and, consequently, not straightforward to take into account. Hence when using modelling approaches based on measured input-output data, such information is already inherently contained in the meas-

ured signals. State-space averaging models taking parasitic elements into account can be found in, e.g. (Ivan et al. 2006; Davoudi and Jatskevich 2007; Xie et al. 2010) and an approach based on measured input-output data, making use of a Hammerstein structure can be found in (Alonge et al. 2007). However, these modelling approaches consider an invariant, resistive output load only, while, the proposed SDP modelling approach considered here makes explicit use of output current measurements, which allows coverage of a wider operating range of the converter.

5.1 The DC-DC boost converter

As mentioned above, the DC-DC boost converter steps-up a DC voltage to a higher DC output voltage, i.e. the purpose is comparable to a transformer in AC. The simplified schematic, i.e. without parasitic elements etc., of a DC-DC boost converter is shown in Figure 5.1, where L, C, and S denote the inductor, capacitor and switch, respectively, which is driven by a pulse-width modulated (PWM) voltage with duty-cycle, denoted d. The quantities V_i, V_o, v_L, and v_C denote the input voltage, output voltage, voltage drop across the inductor and voltage drop across the capacitor, respectively. The inductor current, capacitor current and load, i.e. output current, are denoted by i_L, i_C and i_R, respectively.

In Figure 5.1, the load is represented by a resistor, denoted R. In order to perform the proposed system identification method and in order to consider different load scenarios, the load R is realised as depicted in Figure 5.2, where the inside of the dashed box symbolises the inside of the load resistor R, shown in Figure 5.1. Essentially, the load R, as shown in Figure 5.2, is a voltage controlled current sink, cf. (Horowitz and Hill 1989, Chap. 4). The operational amplifier (OP-amp) adjusts the equivalent resistance of the transistor, by applying an

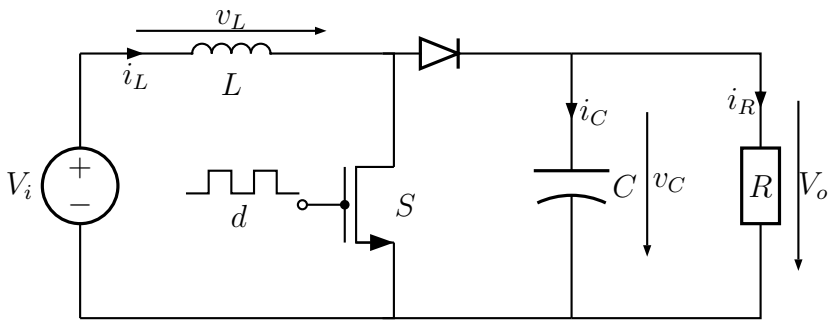


Figure 5.1: Simplified schematic of the DC-DC boost converter

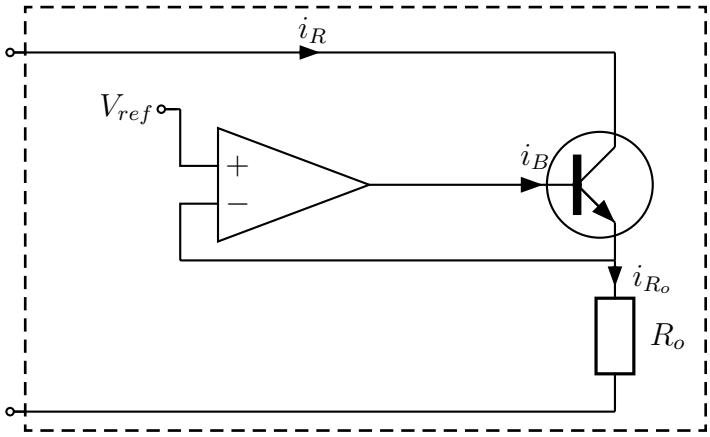


Figure 5.2: Schematic of the load represented by R (dashed box)

adequate transistor base current i_B, such that the voltage drop across the shunt resistor R_o is equal to the load reference voltage denoted V_{ref}. From Ohm's law, it follows that $i_{R_o} = \frac{V_{ref}}{R_o}$ and since, by choosing an adequate transistor, $i_{R_o} = i_R + i_B$ with $i_R \gg i_B$, it can be said that $i_R = i_{R_o}$. Consequently, the current drawn from the converter can be determined by the load reference voltage V_{ref} via the relationship $i_R = \frac{V_{ref}}{R_o}$. Hence realising the load in this way, provides the opportunity for considering various load scenarios.

5.1.1 Operational principle

Here, the operational principle of the DC-DC boost converter is briefly summarized. Detailed information can be found in, e.g. (Mohan et al. 1995; Erickson and Maksimovic 2001; Hitzemann 2009).

The DC-DC boost converter, essentially, consists of three components, as shown in Figure 5.1, namely, the inductor, capacitor and the switch. Both the inductor and capacitor are capable of storing energy. The switch, realised as a N-channel MOSFET, is driven by a PWM voltage of period T_s and duty-cycle d, defined as

$$d = \frac{T_{on}}{T_s} \tag{5.1}$$

where T_{on} denotes the time interval within the period T_s when the PWM voltage is high, which causes the switch to conduct. Consequently, T_{off} denotes the time interval when the PWM voltage is low, which causes the switch not to conduct. Hence the switching period is defined to be

$$T_s = T_{on} + T_{off} \tag{5.2}$$

Now, consider the time interval of a PWM period T_s when the switch is conducting, i.e. during T_{on}. This effectively means that the switch is short-circuited so that the inductor only is charged by the input supply source and the capacitor only supplies the load. The diode, however, ensures that the capacitor is not short-circuited and the current i_C is only able to flow to the load. This basically separates the circuit in two parts as schematically shown in Figure 5.3. The inductor current i_L increases by

$$i_L = \frac{1}{L} \int_0^{T_{on}} v_L \, dt \tag{5.3}$$

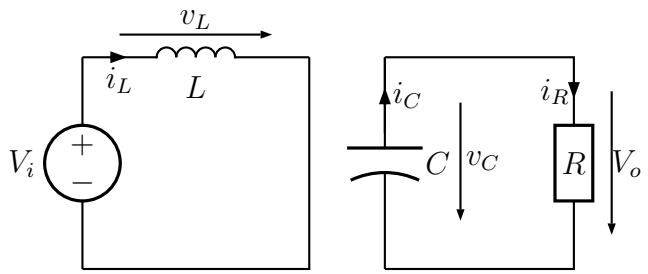


Figure 5.3: DC-DC boost converter schematic when the switch is conducting, i.e. during T_{on}.

where, $v_L = V_i$. The increasing current builds the magnetic field of the inductor, where energy is stored (Griffiths 1999). Simultaneously, the output voltage decreases by

$$v_C = \frac{1}{C} \int_0^{T_{on}} -i_R \, dt \tag{5.4}$$

which is caused by drawing current from the capacitor. Note that when taking Figure 5.3 into account, $v_C = V_o$ and $i_C = i_R$.

Next, consider the time interval of the PWM period T_s when the switch is not conducting, i.e. during T_{off}. This effectively means that the switch is 'removed', so that the circuit in Figure 5.1 can be re-drawn as shown in Figure 5.4. The charged inductor transfers its stored energy to the capacitor and the

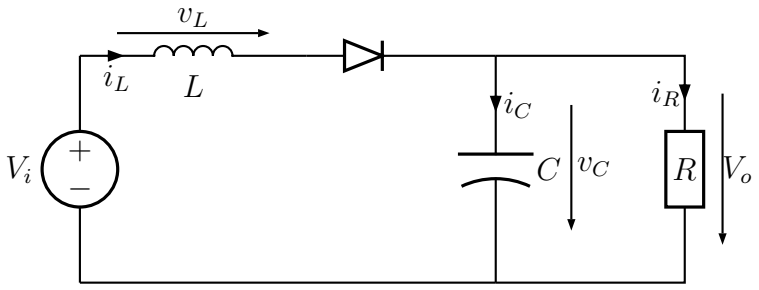


Figure 5.4: DC-DC boost converter schematic when the switch is not conducting, i.e. during T_{off}.

load. Hence the inductor current decreases by

$$i_L = \frac{1}{L} \iint_{T_{on}}^{T_{on}+T_{off}} v_L \, dt \tag{5.5}$$

where the voltage drop across the inductor can be approximated by $v_L \approx V_i - V_o$, which is negative, since in a boost converter, $V_o > V_i$. Moreover, the current i_L can only flow through the diode in the direction towards the capacitor and load, which means that, $i_L \geq 0$. Consequently, if the next period begins before the inductor current i_L reaches zero, i.e. $i_L > 0 \ \forall t$, the inductor conducts continuously and the converter is said to operate in continuous conduction mode (CCM). On the contrary, if the inductor current settles to zero and, subsequently, remains there until the end of the period, the converter is said to operate in DCM, which is considered here. In Figure 5.5, the inductor current waveform for continuous and discontinuous operation is illustrated, where T_z denotes the time interval of T_{off} where the inductor current is zero. Since, during T_{off}, the capacitor is charged,

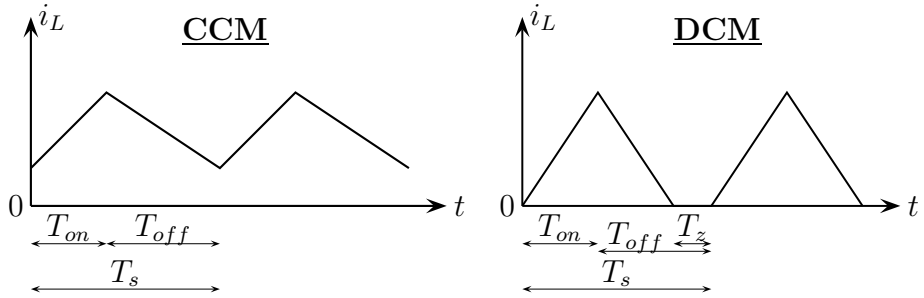


Figure 5.5: Idealised inductor current in continuous (CCM) and discontinuous (DCM) conduction mode operation.

i.e. current flows into the capacitor, the voltage across the capacitor v_C increases by

$$v_C = \frac{1}{C} \iint_{T_{on}}^{T_s} i_C \, dt \tag{5.6}$$

with $v_C \cong V_o$ and taking (5.4) into account, an output voltage ripple is imposed.

This ripple is largely determined by the size of the capacitor and here, the capacitor is chosen large enough so that the ripple may be neglected.

In summary, when considering the input voltage V_i to be constant, during T_{on} the inductor is charged while the capacitor only supplies the load and, subsequently, during T_{off} the charged inductor transfers its energy to the capacitor and the load, supported by the input source. Consequently, when adopting the law of energy conservation, it can be concluded that the output voltage V_o can be controlled by changing the duty-cycle d of the PWM voltage, i.e. adapting T_{on} and T_{off}, respectively.

This means, when referring the boost converter as a system with a system output y and a system input u, that, and in the remainder of this Thesis, for the sake of simplicity, $y \mathrel{\widehat{=}} V_o$ and $u \mathrel{\widehat{=}} d$.

5.1.2 Converter set-up

The set-up of the prototype converter used for laboratory experiments is as follows: $V_i = 5\,\mathrm{V}$, $L = 745\,\mu\mathrm{H}$ with inherent DC series resistance of $DCR \approx 1.3$ and $C = 1000\,\mu\mathrm{F}$. The N-channel MOSFET used, realising the switch S, is the IRLB8748PbF, which on-resistance $R_{DS_{on}} \ll DCR$, hence negligible.

For DCM operation, the switching period, which is also equivalent to the sampling interval, is chosen to be $T_s = 1\,\mathrm{ms}$. In order to generate the PWM voltage signal, the load reference voltage V_{ref} and to acquire the required measurements, the dSPACE MicroAutobox DS1401 is used.

The maximal output voltage V_o is chosen to be $V_o = 20\,\mathrm{V}$, hence the output voltage is defined to be in the range

$$\{V_o \in \mathbb{R} \mid 5\,\mathrm{V} \leq V_o \leq 20\,\mathrm{V}\} \tag{5.7}$$

The maximum current, which can be delivered by the power supply is limited, hence the value $i_L = 2\,\mathrm{A}$ cannot be exceeded. Consequently, the output current is defined to be in the range

$$\{i_R \in \mathbb{R} \mid 40\,\mathrm{mA} \leq i_R \leq 140\,\mathrm{mA}\} \tag{5.8}$$

Furthermore, the duty-cycle of the PWM voltage signal can only vary between 0 % and 100 %, hence d is defined to be in the per-unit range

$$\{d \in \mathbb{R} \mid 0 \leq d \leq 1\} \tag{5.9}$$

Regarding the realisation of the load, as shown in Figure 5.2, the shunt resistor is chosen to be $R_o = 10$, the Op-amp is the LM358N and the transistor is a TIP110 with, according to the datasheet, a typical DC current gain of $h_{FE} = 1000$, so that the requirement $i_B \ll i_R$ is fulfilled.

5.2 State-dependent parameter modelling

The proposed modelling approach is based on measured signals. As mentioned in Section 5.1.1, the system input is given by the duty-cycle of the PWM voltage signal while the system output is given by the output voltage. However, there is an additional measurable signal, namely, the output current. Consequently, modelling the system requires the input d, output V_o as well as the output current i_R to be taken into account.

Naturally, the system input is a signal applied to the system, hence known, while the output is the measured response to that input and the model is required to accurately replicate the system output based on this input. The out-

put current, however, is dependent on the load applied, which may be unknown, but, nevertheless, influences the system output response.

In order to deal with this issue, the output response of a staircase input is obtained while the output current drawn is kept at a constant value, which is possible due to the realisation of the load according to Figure 5.2. This is repeated for various output current values in order to cover the entire operating range of the system. Here, eleven equally spaced output current levels, covering the defined output current range (5.8), are used, i.e. $i_R = 40\,\text{mA},\ 50\,\text{mA},\ 60\,\text{mA},\ \ldots\ ,140\,\text{mA}$. The height of the steps comprising the staircase input are chosen such that the output response increases by a value between approx. 0.5 V and 1 V. An exemplary staircase response where the output current value is kept constant at $i_R = 100\,\text{mA}$ throughout, is shown in Figure 5.6. In Figure 5.7, a single, yet representative, step response taken from Figure 5.6, is shown. Since the model is required to be able to explain the steady-state and dynamic behaviour of the system appropriately and the fact that this information is contained in the staircase responses, the modelling approach is based on the data obtained in this way.

Remark 5.2.1. *Since the output current is kept constant, the dynamics introduced by rapidly changing output currents are not considered. It is assumed that these are mainly 'compensated' by the associated change of operating point and the remaining dynamics are negligible. However, this issue may require further consideration and is left as an open problem.*

5.2.1 Steady-state behaviour

Having acquired the staircase responses which covers the entire range of operation, the steady-state behaviour is examined and, is subsequently modelled.

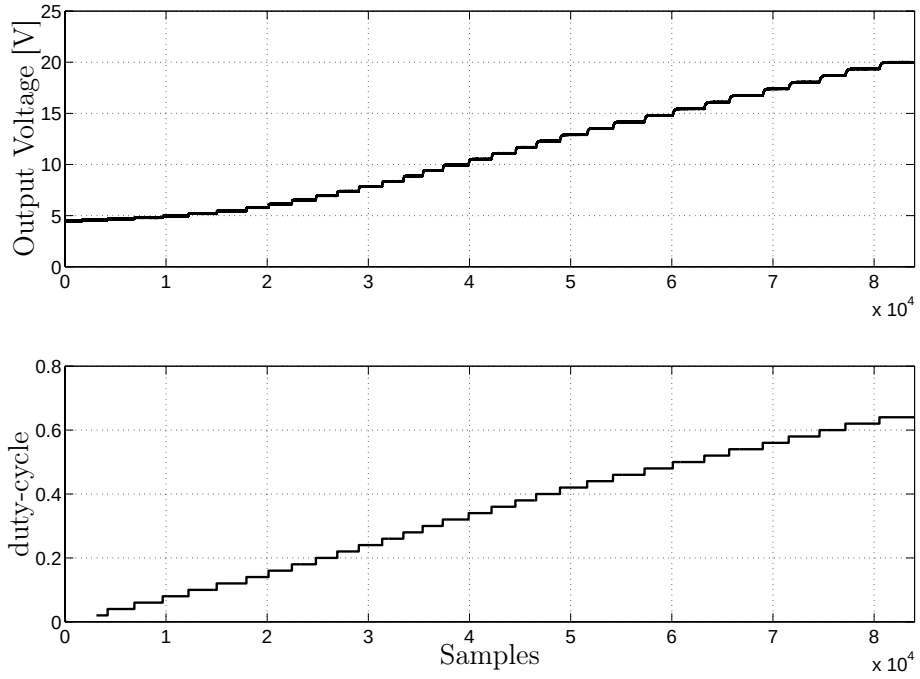


Figure 5.6: Measured output voltage response (upper) to the staircase input (lower) with constant output current value of $i_R = 100\,\mathrm{mA}$

The steady-state behaviour, directly obtained from the measurements, i.e. the staircase responses for constant output current values $i_R = 40\,\mathrm{mA}$, $50\,\mathrm{mA}$, $\ldots$, $140\,\mathrm{mA}$, are shown in Figure 5.8. Consequently, each trace corresponds to a constant output current value in ascending order from left to right. The steady-state behaviour for each output current value is considered individually and modelled by fitting polynomials of the form

$$y_{i,\infty}(u_\infty) = \sum_{j=1}^{4} \beta_{i,j}\ u_\infty^{4-j} \quad \forall i = 1, 2, \ldots, 11 \tag{5.10}$$

where the subscript ∞ indicates steady-state and $\beta_{i,j} \in \mathbb{R}$ denote corresponding coefficients.

For all the i output current values, polynomials comprising the four coefficients $\beta_{i,j} \in \mathbb{R}$ are identified. Since the i polynomials are all of the same

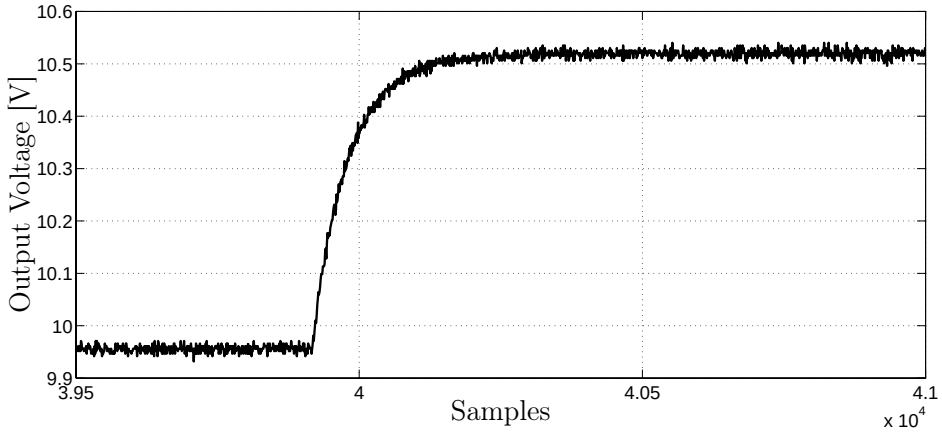


Figure 5.7: Single step response of the staircase responses.

order, the coefficients can be plotted against the output current and, consequently, being realised themselves as functions of the output current, which are, again, chosen to be of a polynomial form

$$\beta_i(i_R) = \sum_{j=1}^{4} \gamma_{i,j}\ i_R^{4-j} \quad \text{with} \quad i = 1, 2, 3, 4 \tag{5.11}$$

where $\gamma_{i,j} \in \mathbb{R}$ denote constant coefficients. The coefficients $\beta_{i,j}$, obtained from (5.10) (solid line) and $\beta_i(i_R)$ being a function of the output current (5.11) (dashed line), are shown in Figure 5.9.

Combining (5.10) with (5.11), the overall steady-state behaviour is characterised by

$$y_\infty(i_R, u_\infty) = \sum_{j=1}^{4} \Big(\beta_j(i_R)\ u_\infty^{4-j} \tag{5.12}$$

The steady-state behaviour directly obtained from measurements and modelled by fitting polynomials (5.10) (solid lines), compared to the steady-state characteristic modelled by (5.12) (dashed line), are shown in Figure 5.10.

Remark 5.2.2. *The order of the polynomials* (5.10) *and* (5.11) *are found by evaluating the criteria chosen to be the mean integral of absolute errors. Essen-*

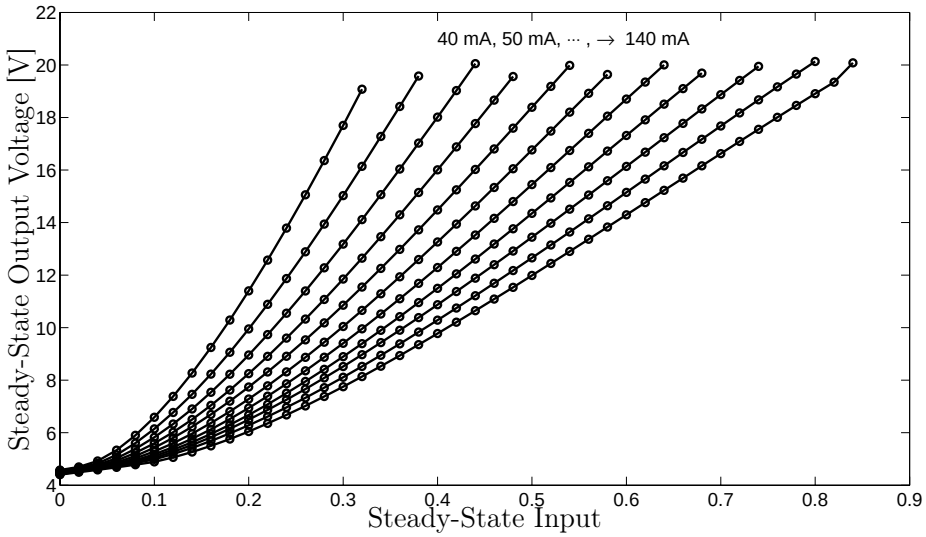


Figure 5.8: Measured steady-state behaviour considered at constant output current values starting at $i_R = 40\,\text{mA}$ increasing in steps of 10 mA up to $i_R = 140\,\text{mA}$, from left to right.

tially, the order, where the difference of the mean integral of errors to the next higher order is sufficiently small, is selected. This means, that the improvement to the next higher order is marginal.

5.2.2 Dynamic behaviour

Having obtained the steady-state behaviour, the dynamics of the system is considered now. The dynamic characteristics are identified by using the individual steps, as shown in Figure 5.7, of the staircase responses, whereby, initially, the staircase responses corresponding to the individual output current values are examined separately. Since the dynamics are of interest, the time-constants of the staircase step responses are required to be obtained. For this reason, consider the linear, discrete-time, first order system model representation

$$y_k = -a_1\ y_{k-1} + b_1\ u_{k-1} \tag{5.13}$$

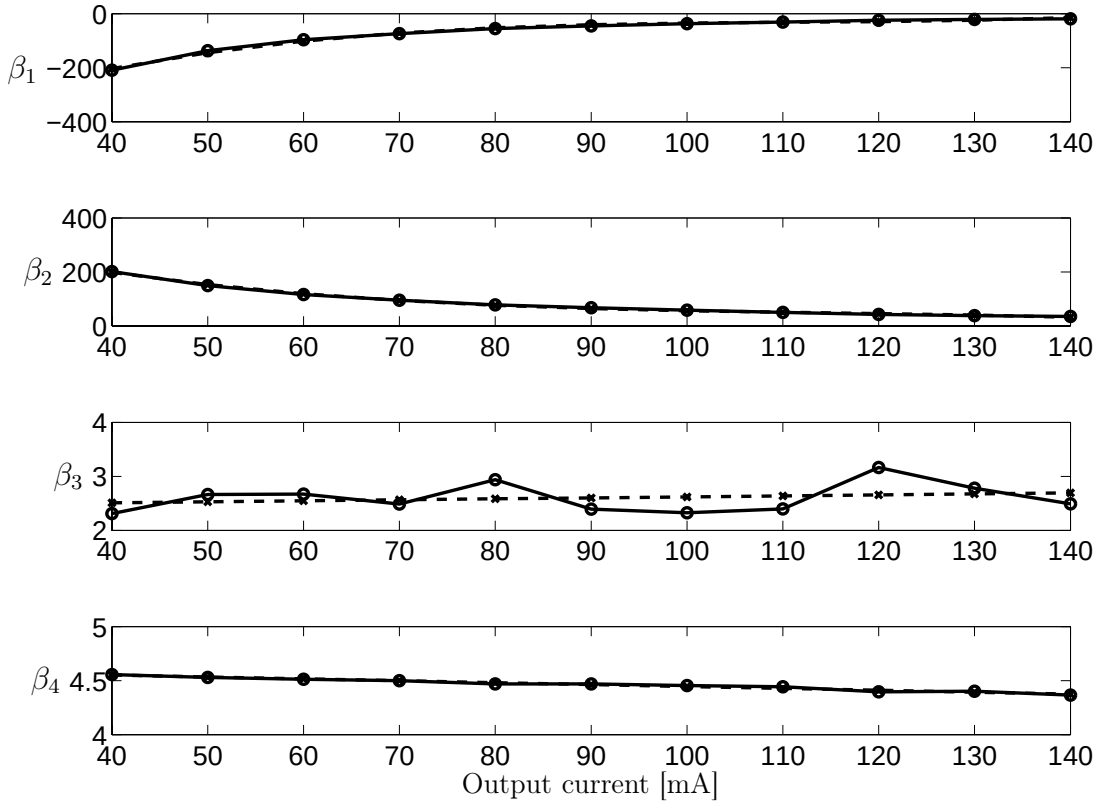


Figure 5.9: Parameters β_i obtained as a function of the output current (dashed line) and obtained directly from considering the individual steady-state behaviours (solid line).

where $a_1 \in \mathbb{R}$ and $b_1 \in \mathbb{R}$ denote model parameters, respectively. In particular, the parameter a_1 is of interest since this parameter relates directly to the equivalent system time-constant at the considered operating point.

In Figure 5.11, the identified model parameter a_1, against the output voltage is presented, where each trace corresponds to a fixed output current value $i_R = 40\,\text{mA},\ 50\,\text{mA},\ \ldots\ ,140\,\text{mA}$, similar to Figure 5.8. Since the discrete-time model parameter a_1 relates to the time-constant by the mapping

$$\tau = \frac{-T_s}{\ln(-a_1)} \tag{5.14}$$

where $\tau \in \mathbb{R}$ denotes the equivalent time-constant of the system at a certain

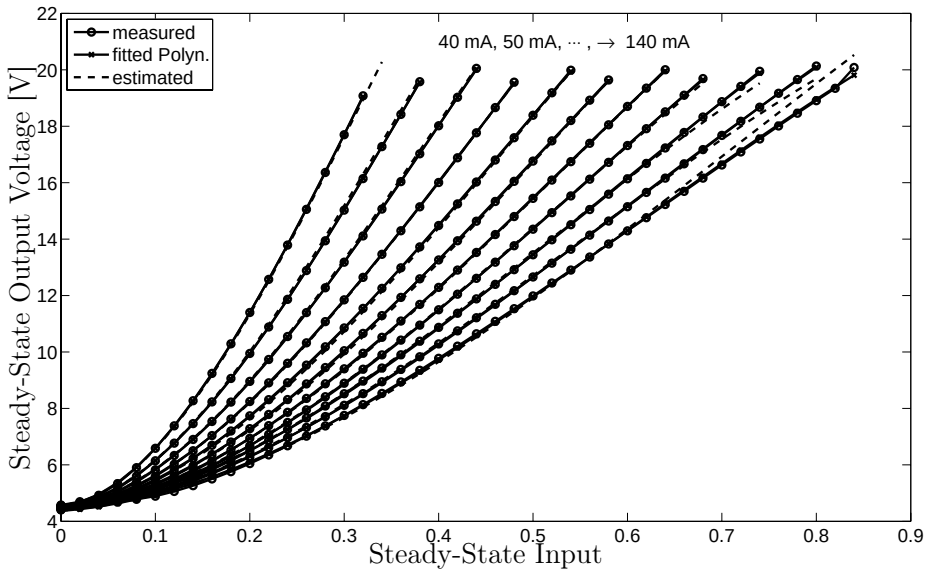


Figure 5.10: Measured steady-state behaviour and fitted individual polynomials corresponding to output current value (solid lines), as well as overall modelled steady-state behaviour (dashed line).

operating point, a linear relationship of the time-constant against the output voltage can be observed in Figure 5.13 (solid line). Consequently, the time-constant can be approximated by first order polynomials of the form

$$\tau_i = \alpha_{i,1}\ y_k + \alpha_{i,2} \quad i = 1, 2, \ldots, 11 \tag{5.15}$$

where each output current value is considered individually, analogously to (5.10). Subsequently, in the same manner as in Section 5.2.1, the coefficients $\alpha_{i,1}$ and $\alpha_{i,2}$, respectively, obtained from (5.15), are considered to be output current dependent and, again, of polynomial form

$$\alpha_j(i_R) = \sum_{l=1}^{4} \Big(\eta_{j,l}\ i_R^{4-l} \quad j = 1, 2 \tag{5.16}$$

where $\eta_{j,l} \in \mathbb{R}$ denote constant coefficients. In Figure 5.12 the coefficients $\alpha_{i,1}$

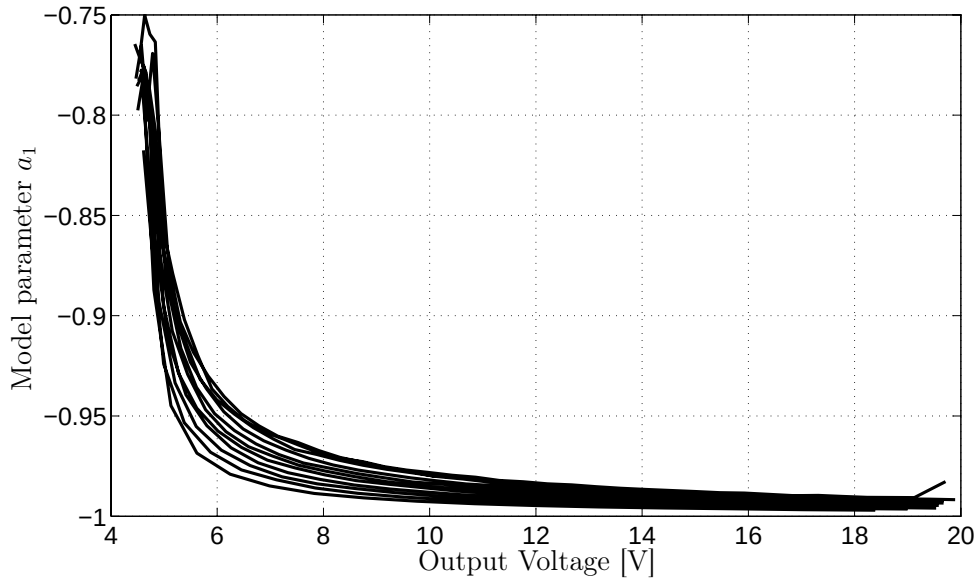


Figure 5.11: Identified linear model parameter a_1 values against output voltage, where each trace corresponds to an output current value of $i_R = 40\,\text{mA},\ 50\,\text{mA},\ \ldots\ , 140\,\text{mA}$.

and $\alpha_{i,2}$ obtained from (5.15) (solid line) as well as $\alpha_j(i_R)$ according to (5.16) (dashed line), are shown. Substituting (5.16) into (5.15), the overall dynamics, represented by the equivalent time-constant, is approximated by

$$\tau(y_{k-1}, i_{R,k}) = \alpha_1(i_{R,k})\ y_{k-1} + \alpha_2(i_{R,k}) \tag{5.17}$$

and shown in Figure 5.13 (dashed line).

When comparing Figures 5.11 and 5.13, it can be observed that in the discrete-time domain, i.e. Figure 5.11, the differences of the traces seem to be marginal, in particular with increasing output voltage. On the contrary, in Figure 5.13, i.e. when mapping in the continuous time-domain, it can be observed that the spread of the traces, in fact increase with increasing output voltage so that at the widest spread, the longest equivalent time-constant of the slowest mode is approximately three times that of the fastest mode.

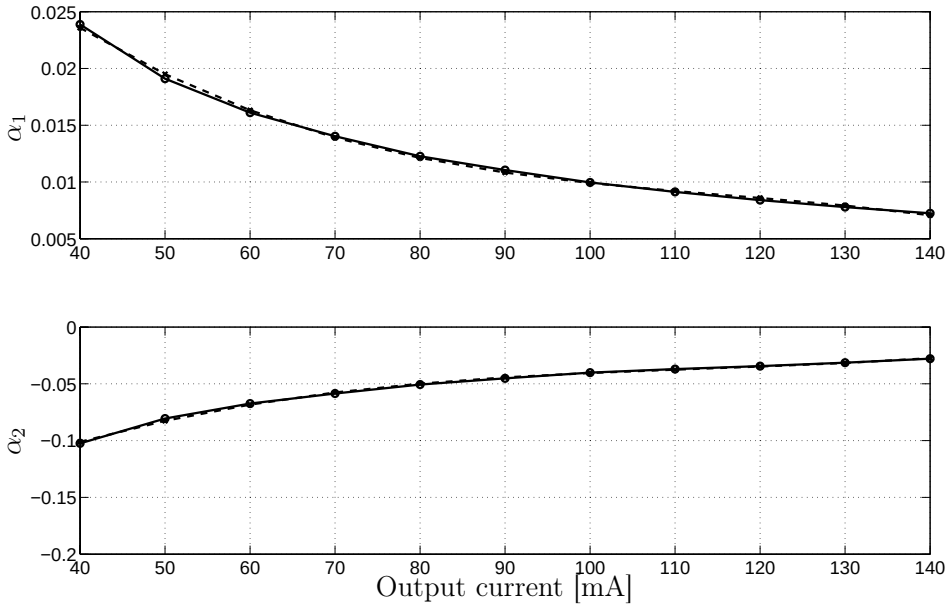


Figure 5.12: Parameters α_j obtained as a function of the output current (dashed line) and directly obtained from considering the dynamic behaviours corresponding to the different output current values.

Also, when drawing a lower output current from the converter, the equivalent time-constant is longer. This can be explained by (5.4), which implies that the output voltage changes with the integral of the output current drawn.

5.2.3 Obtaining the SDP model

Finally, the system is sufficiently characterised by its steady-state (5.12) and dynamic (5.17) behaviour so that the SDP model can be obtained.

From Figure 5.7, it can be deduced that a first order model is an appropriate choice, hence

$$y_k = -a_{1,k}\ y_{k-1} + b_{1,k}\ u_{k-1} \tag{5.18}$$

where the parameters $a_{1,k}$ and $b_{1,k}$ are required to be obtained.

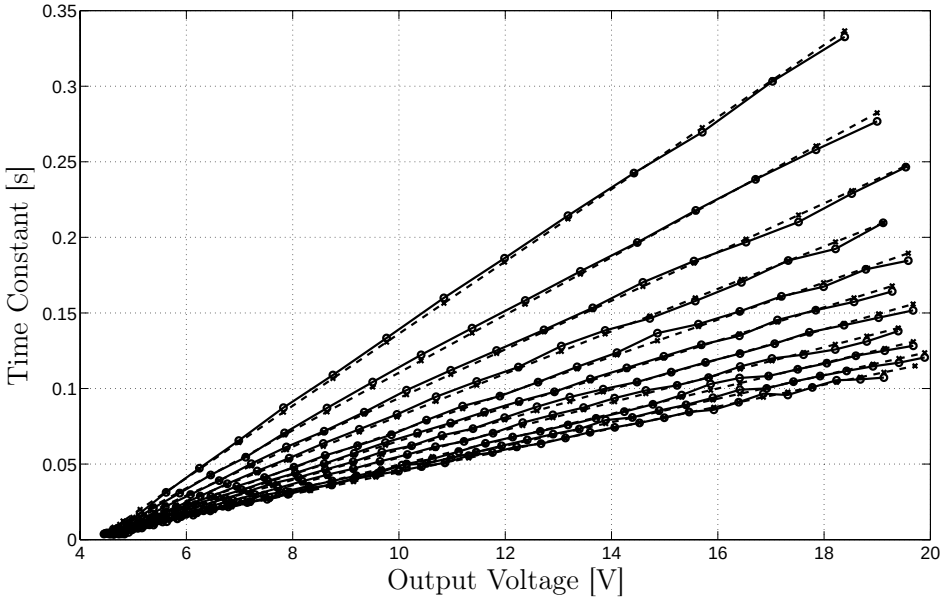


Figure 5.13: Time constant obtained from steps of staircase responses directly (solid line) and modelled (dashed line), against the output voltage. Each trace corresponds to a fixed output current value of $i_R = 40\,\text{mA}, 50\,\text{mA}, \ldots, 140\,\text{mA}$, from upper to lower.

Initially, consider the parameter $a_{1,k}$. Analogously to (5.13), $a_{1,k}$ is used in order to model the dynamic behaviour. Mapping (5.17) back into the discrete-time domain, cf. (5.14), $a_{1,k}$ is given by

$$a_{1,k} = -e^{\frac{-T_s}{\tau(y_{k-1}, i_{R,k})}} \tag{5.19}$$

The model parameter $b_{1,k}$, however, is used in order to satisfy the steady-state behaviour (5.12). Considering (5.18) in steady-state and subsequently solving for $b_{1,\infty}$, yields

$$b_{1,\infty} = \frac{y_\infty(1 + a_{1,\infty})}{u_\infty} \tag{5.20}$$

when $u_\infty = y_\infty^{-1}$, i.e. the steady-state input is equal to the inverse function of the

steady-state output, $b_{1,k}$ can be approximated by

$$b_{1,k} = \frac{y_{k-1}\ (1 + a_{1,k})}{y_{\infty}^{-1}(y_{k-1}, i_{R,k})} \tag{5.21}$$

where, taking (5.12) into account, the inverse of a cubic polynomial is required, which is obtained, by following (Nickalls 2006), to be

$$y_{\infty}^{-1}(y_{\infty}, i_R) = 2\sqrt{\left(-\frac{w}{3}\right)}\cos\left[\frac{1}{3}\cos^{-1}\quad\frac{3q}{2w}\sqrt{\left(-\frac{3}{w}\right)}\right]\left(-\frac{2\pi}{3}\right]\left(-\frac{\beta_2(i_R)}{3\beta_1(i_R)}\right. \tag{5.22a}$$

with

$$w = \frac{3\beta_1(i_R)\beta_3(i_R) - \beta_2^2(i_R)}{3\beta_1^2(i_R)} \tag{5.22b}$$

and

$$q = \frac{2\beta_2^3(i_R) - 9\prod_{i=1}^{3}\beta_i(i_R) + 27\beta_1^2(i_R)\ (\beta_4(i_R) - y_{\infty})}{27\beta_1^3(i_R)} \tag{5.22c}$$

Now, having obtained the SDP model, some comments regarding the model can be made:

- In the proposed modelling approach, use is made of polynomials describing the steady-state and dynamic behaviour. This inflicts certain issues, such as the fact that the defined operating range must not be exceeded since, in particular, if higher order polynomials are used, the behaviour outside the considered operating range may change significantly. Further details on the drawbacks of using polynomials in system identification can be found in (Nelles 2001, Chap. 18). On the other hand, the identification of the coefficients is straightforward since polynomials are linear w.r.t the coefficients and, consequently, well known system identification methods, such as the least-squares algorithm, can be used.

- The steady-state behaviour (5.12), as used in (5.20) and (5.21), respectively,

as well as the dynamic behaviour (5.17) are dependent on y_{k-1} instead of y_k.

- Considering (5.17), each step is chosen to be of a height, as mentioned above, between 0.5 V and 1 V, so that the obtained time-constant of each individual step applies to this output voltage range, hence if the sampling interval is sufficiently short, the difference between y_k and y_{k-1} is within this range. Consequently, using y_k, y_{k-1} or even y_{k+1}[1] is, from this point of view, of minor relevance. However, in the light of a subsequent application to a SDP-PIP controller, it may be advantageous to make use of y_{k-1}, as depicted in Chapter 6.
- Regarding the use of the steady-state behaviour in the model parameter $b_{1,k}$, again, the actual measured data points are distinct and based on the height of the steps in the staircase responses, where interpolation is used in between so that, if the difference between y_k and y_{k-1} is sufficiently small, making use of y_{k-1} can be seen as an appropriate approximation. Additionally, in steady-state, it can be said that $y_k = y_{k-1}$.

Experimental model validation

In Figure 5.14, the response of the model and the measured response of the converter (upper) is shown for an arbitrarily varying duty-cycle (lower) and an arbitrarily varying output current (middle).

It can be observed that the model and measured output voltage response is almost identical. A mismatch is visible at high output voltage, in particular, if the output current is high as well. This becomes apparent when comparing the

[1] which is rather of a theoretical nature than of practical relevance

peak between approximately $t \approx 12\,\mathrm{s} - 15\,\mathrm{s}$ where the output current is high and the peak between approximately $t \approx 75\,\mathrm{s} - 77\,\mathrm{s}$ where the output current is in the midrange. The mismatch in the latter peak is clearly less than in the first considered peak. Furthermore, this observation is also reflected in Figure 5.10 where at high output voltage, a mismatch at high output current is observed.

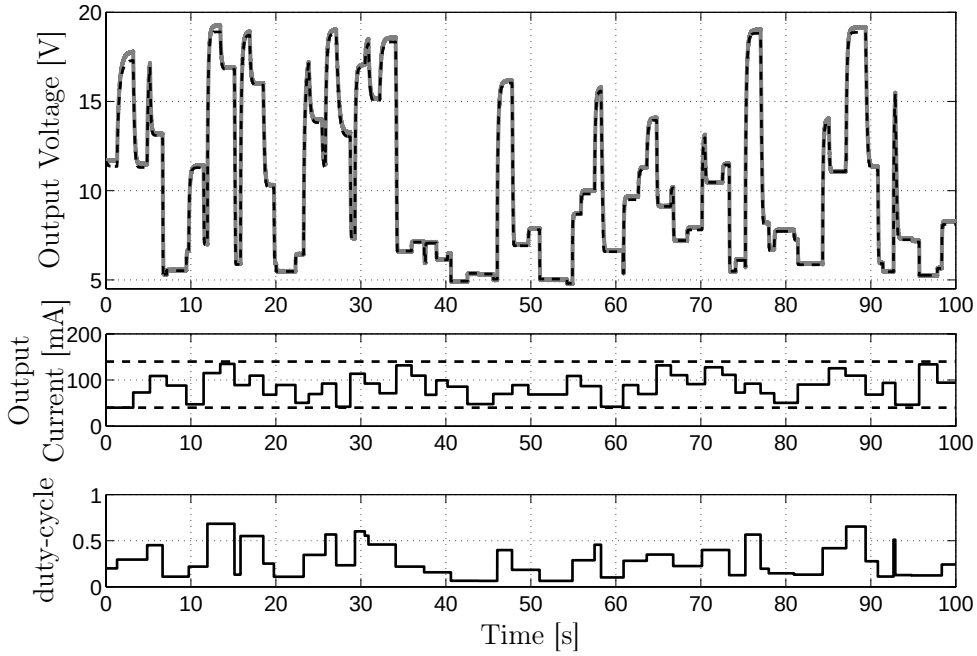


Figure 5.14: Upper: measured output voltage (solid line) and output voltage response of the model (dashed line) to arbitrarily varying system input and output current. Middle: arbitrarily varying output current drawn from the converter (solid line) within the defined range (dashed lines). Lower: arbitrarily varying system input.

5.3 Comparison of SDP with HBS modelling approach

In this Section, the proposed state-dependent parameter model (5.18), in conjunction with (5.19) and (5.21), respectively, is compared to a modelling approach based on a Hammerstein-bilinear structured (HBS) model. The HBS model is briefly introduced in the following, as presented in (Larkowski et al. 2013).

Generally, the HBS consists of a Hammerstein term, i.e. static nonlinearity, denoted $f(\cdot)$, followed by a bilinear dynamic subsystem, where the output of the static nonlinearity, denoted v_k, is the input to the bilinear subsystem, hence the system input u_k is transformed by $f(\cdot)$ in order to form the input to the bilinear subsystem, as shown in Figure 5.15. Consequently, the steady-state behaviour is modelled by the static nonlinearity, while the dynamics are modelled by the bilinear subsystem. The HBS model can be described by the following nonlinear

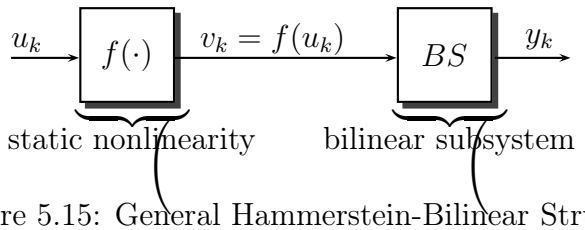


Figure 5.15: General Hammerstein-Bilinear Structure

discrete-time difference equation, in a general form, i.e.

$$v_k = f(u_k) \tag{5.23a}$$

$$\begin{aligned} y_k = &- \sum_{i=1}^{n_a} a_i \; y_{k-i} + \sum_{j=1}^{n_b} b_j \; v_{k-j} \\ &+ \sum_{i=1}^{n_a} \sum_{j=1}^{n_b} \eta_{ij} \; y_{k-i} \; v_{k-j} + c \end{aligned} \tag{5.23b}$$

where a_i, b_j, η_{ij} and $c \in \mathbb{R}$ denote model parameters, respectively.

5.3.1 Hammerstein-bilinear modelling approach

The HBS modelling approach is also based on measured input-output data. However, differently to the SDP modelling approach presented in Section 5.2, these data are not obtained from staircase responses, but rather by applying an arbitrarily varying input signal, which causes excitation of the system throughout the entire output voltage range, while the output current is kept at fixed constant levels of $i_R = 40\,\text{mA}, 50\,\text{mA}, \ldots, 140\,\text{mA}$, hence eleven data sets are acquired. In Figure 5.16, a representative data set where $i_R = 100\,\text{mA}$, is presented. For each

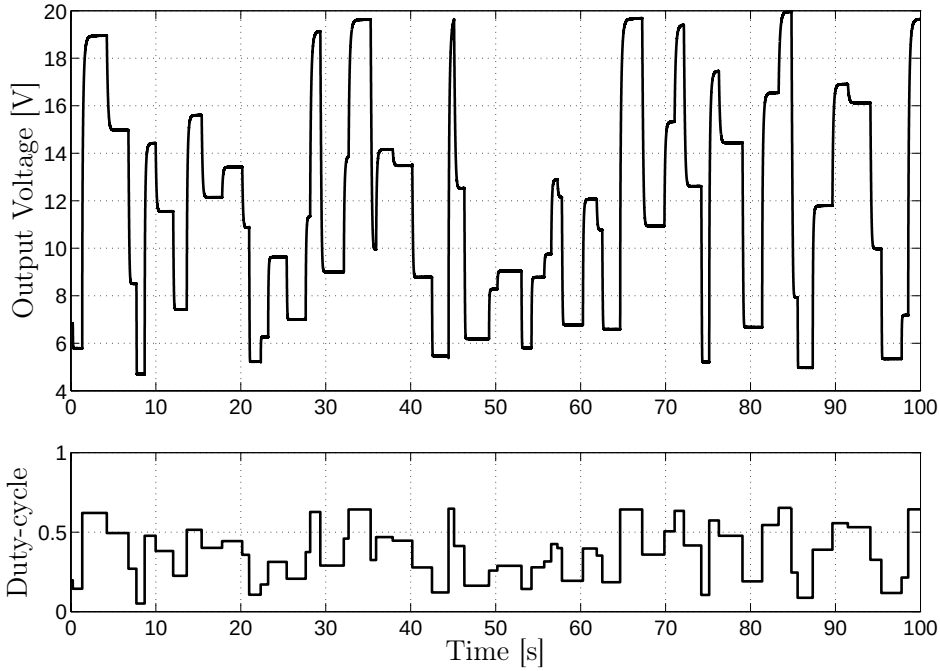


Figure 5.16: Measured output voltage response (upper) to an arbitrarily varying input (lower), such that the entire output voltage range is covered where the output current is kept constant throughout at $i_R = 100\,\text{mA}$.

individual data set, a separate HBS model of the form (5.23) is obtained. The static nonlinearity (5.23a) is chosen to be of polynomial form, i.e.

$$f(u_k) = \alpha_0 + \alpha_1\ u_k + \alpha_2\ u_k^2 + \alpha_3\ u_k^3 + \alpha_4\ u_k^4 \tag{5.24}$$

where $\alpha_i \in \mathbb{R}$ denote invariant coefficients. The HBS, given by (5.23) - (5.24), is nonlinear w.r.t. the input-output, however, it is bilinear w.r.t. the parametrisation, so that, regarding the parameter estimation task, a so-called bilinear parametrisation method can be used, see (Ljung 1999). It is an iterative method, comprised of two alternating least-squares (LS) algorithms. The first LS algorithm uses the most recent[2] estimate of the set of parameters defining $f(u_k)$ and estimates the parameters of the bilinear subsystem. Subsequently, the second LS algorithm estimates the parameters of $f(u_k)$, by using the estimates defining the bilinear subsystem obtained by the first LS algorithm. This procedure is repeated until either convergence is achieved or a predefined number of iterations is exceeded.

Regarding the HBS system identification of the considered boost converter, here the model order of the obtained eleven HBS 'sub-models', each of which correspond to a distinct output current level, is given by $n_a = n_b = 1$.

The accuracy of these sub-models are quantified by the criteria, namely, the coefficient of determination (Young 2011) and mean integral of absolute errors, respectively,

$$R_T^2 = 100 \left(1 - \frac{\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2}{\|\boldsymbol{y} - E[\boldsymbol{y}]\|_2^2}\right) \tag{5.25a}$$

$$IAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{5.25b}$$

where $\boldsymbol{y}, \hat{\boldsymbol{y}} \in \mathbb{R}^{N \times 1}$ denote the vectors of measured and model output values, respectively, $E[\boldsymbol{y}]$ denotes the expected value of $\boldsymbol{y}$, N the number of samples, whilst the notation $\|\cdot\|_2$ denotes the Euclidean norm.

In Figure 5.17, the mean integral of absolute errors and the coefficients of determination for all the identified sub-models are shown.

[2] or initial values in case of the first iteration

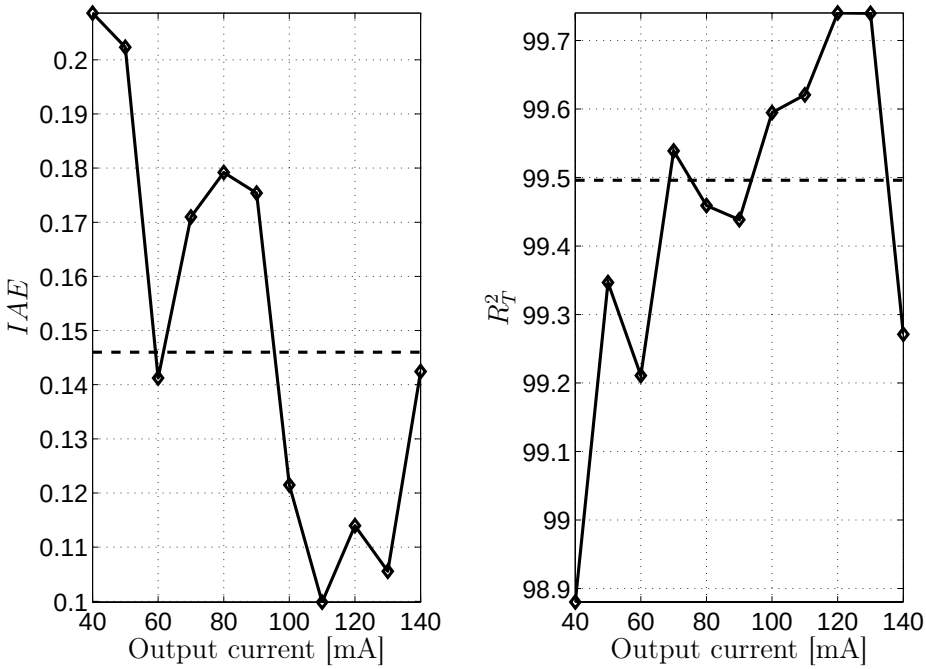


Figure 5.17: Evaluation criteria IAE and R_T^2 of the individual identified HBS sub-models for output current values $i_R = 40\,\text{mA}, 50\,\text{mA}, \ldots, 140\,\text{mA}$. The dashed line denotes the respective average value.

Having obtained the sub-models, in the following, the overall HBS model is now obtained. To create the overall model, the sub-models are required to be combined. A smooth transition is achieved by making use of Gaussian blending, where the means of each membership function is located at the considered, eleven output current values. Consequently, interpolating between the individual sub-models is based on the instantaneous, measured output current value. The Gaussian membership functions, which are normalised such that the aggregated weighting at each point is unity, are shown in Figure 5.18. Finally, the overall

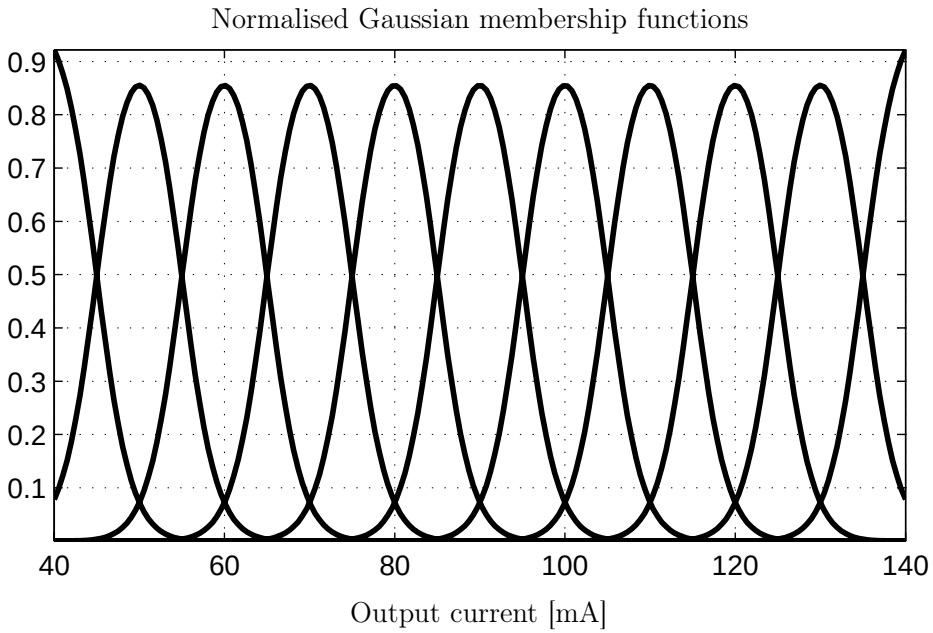


Figure 5.18: Normalised Gaussian membership functions

HBS model is given by

$$\begin{aligned}\hat{y}_k = &- \sum_{i=1}^{n_a} \gamma(\bar{a}_i, i_{R,k})\ \hat{y}_{k-i} + \sum_{j=1}^{n_b} \gamma(\bar{b}_j, i_{R,k})\ v_{k-j} \\ &+ \sum_{i=1}^{n_a}\sum_{j=1}^{n_b} \gamma(\bar{\eta}_{ij}, i_{R,k})\ \hat{y}_{k-i}v_{k-j} + \gamma(\bar{c}, i_{R,k})\end{aligned} \tag{5.26}$$

where $\bar{a}_i$, $\bar{b}_j$, , $\bar{\eta}_{ij}$ and $\bar{c}$ denote parameter sets, which consist of the corresponding identified sub-model parameters. The Gaussian membership function is denoted by $\gamma(\cdot)$ and given by, exemplary shown here for the a_i - parameters,

$$\gamma(\bar{a}_i, i_{R,k}) = \frac{\sum_{l=1}^{11} a_{i,l}\ e^{\frac{-(i_{R,k}-i_{R,l})^2}{2\sigma^2}}}{\sum_{l=1}^{11} e^{\frac{-(i_{R,k}-i_{R,l})^2}{2\sigma^2}}} \tag{5.27}$$

where $i_{R,l}$ denotes the $l-th$ element of the output current set $\{40\,\text{mA}, 50\,\text{mA}, \ldots, 140\,\text{mA}\}$ and, similarly, $a_{i,l}$ denotes the $l-th$ element of the set of parameters $\bar{a}_i$ corresponding to the sub-model, identified at the $l-th$ output current value.

The constant variance of the Gaussian functions is chosen to be $\sigma^2 = 4$, which was a value found by trial and error.

5.3.2 Experimental results and discussion

In Figure 5.19, the measured output voltage response as well as the HBS and SDP model response to an arbitrarily varying duty-cycle and output current are shown. It can be observed that the fit of the SDP response slightly outperforms the HBS model response. This observation is also confirmed by Table 5.1 where the quantified performance criteria (5.25) are presented. However, from visual inspection and supported by the assessment criteria results, the performance of the HBS model is still considered satisfactory.

Criteria	HBS	SDP
IAE	0.14	0.08
R_T^2	99.49	99.86

Table 5.1: Quantified performance assessment criteria comparing the SDP and HBS model performance.

Regarding the HBS compared to the SDP modelling approach, some observations can be made:

- The most apparent difference is that the HBS model is comprised of several sub-models, which are blended by making use of a Gaussian membership function. The SDP model, however, which is also based on data where different output current values are considered, but the 'interpolation' between the distinct output currents is built inherently into the model parameters. Identification of separate sub-models for each operating point is not necessary. It is noted, however, that the experimental data, of the nature and form used for identification of the HBS model, could also be used for iden-

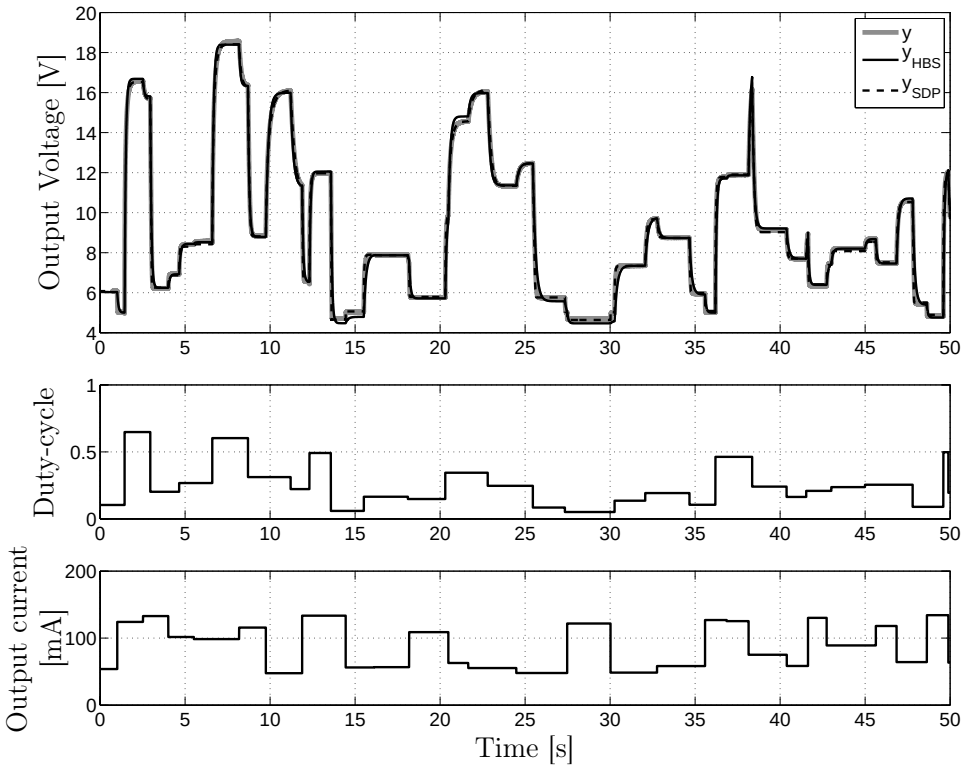


Figure 5.19: Upper: Comparative results of measured output voltage (thick grey line), output voltage response of the HBS model (solid thin line) and SDP model response (dashed line) covering the entire operating range, by using a validation data set. Middle: Corresponding arbitrarily varying duty-cycle. Lower: Corresponding arbitrarily varying output current drawn from the converter.

tification of the SDP model by making use of the 'back-fitting' approach, see e.g. (Taylor et al. 2007a).

- The HBS modelling approach as well as the SDP modelling approach consider the dynamics and steady-state behaviour separately. In the HBS model, the static nonlinearity is dedicated in order to model the steady-state behaviour, while the SDP model dedicates the $b_{1,k}$ parameter for this task. Similarly, the bilinear subsystem of the HBS model takes care of the system dynamics while in the SDP model, this is done by the parameter $a_{1,k}$. Moreover, both approaches make use of polynomials, which allow the

estimation of the required coefficients by making use of the straightforward LS algorithm so that, at this point, no additional burden is introduced.

- In the case of the HBS approach, the sub-models are identified for operating points based on certain output current levels. Consequently, these sub-models can be considered as being optimal (at their corresponding operating point and based on the performance criteria (5.25)) so that it can be said that the achieved performance, as shown in Figure 5.17, is the best achievable performance. Considering the membership functions, shown in Figure 5.18, it can be observed that even if operating at the specific operating points, i.e. when $i_R = 40\,\text{mA}, 50\,\text{mA}, \ldots, 140\,\text{mA}$, the overall, i.e. blended, HBS model is not identical to the optimal sub-model, at this operating point, since both neighbouring membership functions are not zero. As a consequence, the parameters of the overall model differ from the optimal sub-model, hence the performance criteria values shown in Figure 5.17 can be viewed as a form of upper bound that the overall model is able to achieve. This, in turn, allows conclusions to be drawn from Figure 5.17, about the overall performance throughout the operating range. Regarding the HBS model of the boost converter considered here, from Figure 5.17, it is anticipated that the performance in the range of $i_R \approx 100\,\text{mA} - 130\,\text{mA}$ is superior. This is also reflected in Figure 5.19, when comparing the performance of the HBS in the intervals $t \approx 2\,\text{s} - 9\,\text{s}$ where the output current is in this range and $t \approx 30\,\text{s} - 35\,\text{s}$ where the output current is low and a slightly greater mismatch is observed.

5.4 Concluding remarks

In this Chapter, a practical application example has been presented. A modelling approach for a DC-DC boost converter operating in discontinuous conduction mode has been proposed. The modelling approach was based on the SDP framework and makes use of measured input-output data rather than of physical relationships. Staircase system inputs are applied at different output current levels, so that step responses throughout the entire operating range are obtained. Based on this data, the dynamic and steady-state characteristics of the converter were obtained, upon which the SDP model parameters have been identified.

This proposed SDP model has been compared with an alternative model, based on a Hammerstein-bilinear structure. The steady-state characteristic has been captured by the Hammerstein static nonlinearity and the dynamics by the bilinear term. This modelling approach was also based on measured input-output data. Several HBS sub-models, corresponding to certain levels of output currents, have been identified and were subsequently blended by a Gaussian membership function so that an overall model, which covers the entire operating range, was obtained.

Furthermore, the similarities, as well as the differences of these modelling approaches have been highlighted and comparative performance results, based on laboratory based experiments, have been presented.

Chapter 6

SDP-PIP Controller Implementation Results

Based on the elaborated SDP model of the DC-DC boost converter presented in Chapter 5, implementation results of the model based SDP-PIP controller are now presented. In order to demonstrate the efficacy of the proposed SDP-PIP control approach, implementation results are compared with linear PIP control, which is, in the case of a first order model, structurally similar to the widely used non-model based proportional integral (PI) controller, in fact, the linear PIP based on a first order model also consists of a proportional and an integral gain, see e.g. (Taylor et al. 2001). In this regard, these controllers can be seen to be equivalent, however, making use of the linear PIP controller instead of the PI controller, provides a comfortable way of tuning the controller by simply choosing the desired closed-loop poles, to provide the calculation for the controller gains. Moreover, since the SDP-PIP 'replicates' a linear closed-loop system behaviour with closed-loop poles at desired locations, this allows a direct performance comparison.

The output voltage control objectives are twofold. Output voltage regulation when load steps, i.e. the case of output current steps are considered. This

scenario is also mainly considered in the literature. The second objective is to achieve a predefined transient behaviour when output voltage set-point changes are applied. This predefined transient behaviour is determined by the choice of the desired closed-loop pole locations.

Various model based control approaches for DC-DC boost converters can be found in the literature. In (Beccuti et al. 2005, 2007, 2009) an approach can be found, which makes use of a model based predictive control (MPC) strategy, regulating the DC-DC boost converter, while (Geyer et al. 2008) applies the MPC approach to a DC-DC step-down converter. Robust control approaches, which explicitly take model uncertainties into account and represent the possible load variations in the uncertainty formulation, can be found in, e.g. (Cortes et al. 2005; Fadil and Giri 2007; Olalla et al. 2009, 2010, 2011; Sira-Ramirez et al. 2011).

Regarding the DC-DC boost converter particularly operating in DCM, control approaches can be found in, e.g. (Tse and Adams 1990; Qiao and Zhang 2005). These control strategies, are all based on an averaged state-space model and not based on grey-box models as is the SDP model derived in Chapter 5.

6.1 Limitations regarding the output voltage control

As it is mentioned in Section 5.1.1, that the output voltage rate of change is affected by the amount of current drawn from and supplied to the capacitor. The amount drawn from the capacitor is mainly determined by the load and the amount supplied to the capacitor is mainly determined by the duty-cycle. Since the duty-cycle is naturally constrained (5.9), as a consequence, there are limitations on the rate of change of the output voltage, i.e. the maximal slope is

limited. In particular, the maximal achievable output voltage slope is determined by the capacitance, output current, etc. (Wens and Steyaert 2011, Chap. 2), so that when choosing the desired closed-loop pole locations, this is required to be taken into account, cf. (Cunha and Pagano 2002).

Consequently, regarding the SDP-PIP controller implementation, one way to overcome this issue is to place sufficiently 'slow' desired closed-loop poles, such that the duty-cycle does not run into constraints.

6.2 Experimental set-up

The converter set-up used in this Chapter is identical to that of the previous Chapter, as described in Section 5.1.2. Since the load is realised as shown in Figure 5.2, a load scenario is required in order to be able to obtain controller implementation results. Hence the output current i_R, which is drawn from the converter is given by the following first order continuous-time transfer function

$$i_{R,k} = \frac{8}{0.1s + 1} V_{o,k} \tag{6.1}$$

where s denotes the Laplace variable and i_R is obtained in mA while V_o is given in V. This means that the measured output voltage is 'filtered' by the transfer function and the resulting output current is drawn from the converter by applying the corresponding load reference voltage V_{ref}, provided by the DAC of the dSPACE Microautobox. In this manner, a load is replicated such that the output current stays within its defined operating range (5.8).

Additionally, in order to avoid that the input power supply unit runs into its built-in current limiter during the experiment, the duty-cycle is further limited to the range $\{0.05 \leq d \leq 0.9\}$.

6.3 SDP-PIP controller design

Based on the SDP model obtained in Section 5.2.3, the SDP-PIP controller gains are calculated. Since this model (5.18) is of first order and of unit sampling delay, i.e. $n_a = n_b = \tau = 1$, the controller gains $f_{0,k}$ and $K_{I,k}$ are required only. Following Section 4.2.2, these gains are given by

$$f_{0,k} = -\frac{a_{1,k+1} + d_2}{b_{1,k+1}} \tag{6.2a}$$

and

$$K_{I,k} = \frac{1 + d_1 + d_2}{b_{1,k+1}} \tag{6.2b}$$

Note here that the model parameters $a_{1,k+1}$ and $b_{1,k+1}$ are dependent on the sampling instance $k+1$, hence recall the dependency on y_{k-1} used in (5.19) and (5.21), respectively, as discussed in Section 5.2.3. This allows one to conveniently make use of the current output measurement y_k instead of using the predicted value y_{k+1}.

Furthermore, due to the issue mentioned in Section 6.1, it might well be possible that the input runs into constraints, hence, when making use of the integral-of-errors state, integrator windup may occur. Consequently, in order to avoid this, use is made of the SDP-PIP in incremental input form according to Section 4.4.

Moreover, placing the desired closed-loop poles at reasonably slow locations in order to avoid system input saturation as discussed in Section 6.1, leads to pole locations close to the border of the unit circle in the complex plane. Implementation results, where the SDP-PIP controller sampling interval is equal to the switching period, i.e. $T_s = 1\,\text{ms}$, and the desired closed-loop poles, denoted p_1 and p_2, respectively, are both placed at $p_1 = p_2 = 0.70, 0.80, 0.90, 0.95$, are shown

in Figure 6.1. The dashed line shows the desired response while the solid line shows the measured output voltage response to a step set-point change from 7 V to 17 V. Additionally, the corresponding system inputs are shown in Figure 6.2. In both Figures 6.1 and 6.2, the effect of the input saturation is observable. When choosing desired closed-loop poles at locations as slow, i.e. close to the border of the unit circle, as $p_1 = p_2 = 0.95$, the converter is physically able to meet the desired response and, in addition, the SDP-PIP controller is able to accurately track this desired response. Considering Figure 6.2, when fast closed-loop poles are chosen, i.e. $p_1 = p_2 = 0.70$, apparently, the control action is more sensitive to measurement noise, hence more active, than when choosing slow closed-loop poles. The corresponding output voltage, however, does not seem to be more affected by the noise than those where slower closed-loop poles are chosen. This is not surprising when considering, in relation to the switching frequency, the slow equivalent open-loop system time constants, see Section 5.2.2.

6.3.1 SDP-PIP controller sampling interval

In order to increase the range of possible, desired closed-loop pole locations such that the input does not saturate, the controller sampling interval, i.e. the sampling interval upon which the model is based, is required to be increased due to the fact that the pole location in the discrete-time domain depends on this sampling interval (5.19), (5.21).

At this juncture, it is pointed out that changing the discrete-time pole locations via altering the sampling interval, the equivalent continuous-time pole location, i.e. time-constant, remains unchanged. Also, the physical performance constraints discussed in Section 6.1 cannot be affected. The range of numerical values of desired discrete-time closed-loop pole locations, which are physically

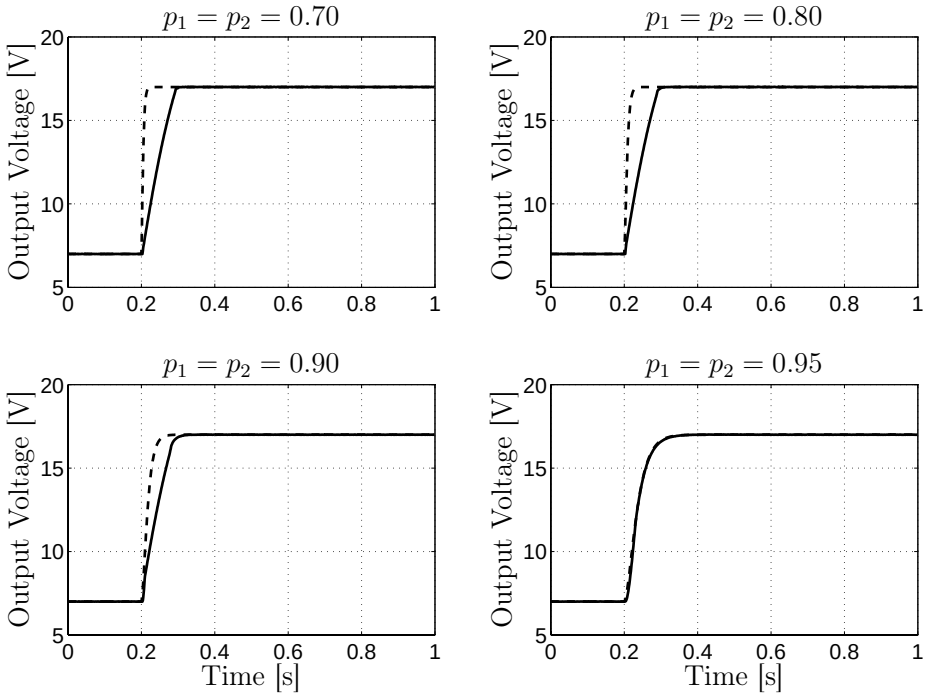


Figure 6.1: Desired output voltage response (dashed line) and measured output voltage (solid line) to a step set-point change from 7 V to 17 V for various desired closed-loop pole locations and a SDP-PIP controller sampling interval of 1 ms.

achievable, can only be increased so that these pole locations are more wide spread and not only concentrated close to the boundary of the unit circle.

The SDP model obtained in Section 5.2 is determined by the steady-state and dynamic behaviour of the system. Since the equivalent time constants are identified in the continuous-time domain and subsequently mapped into the discrete-time domain, it is straightforward to change the controller sampling interval. Consequently, (5.19) becomes

$$a_{1,k} = -e^{\frac{-T_c}{\tau(y_{k-1}, i_{R,k})}} \tag{6.3}$$

where T_c denotes the controller sampling interval. Choosing $T_c = 5\,\text{ms}$, while the switching period remains at $T_s = 1\,\text{ms}$, the controller 'executes' every fifth

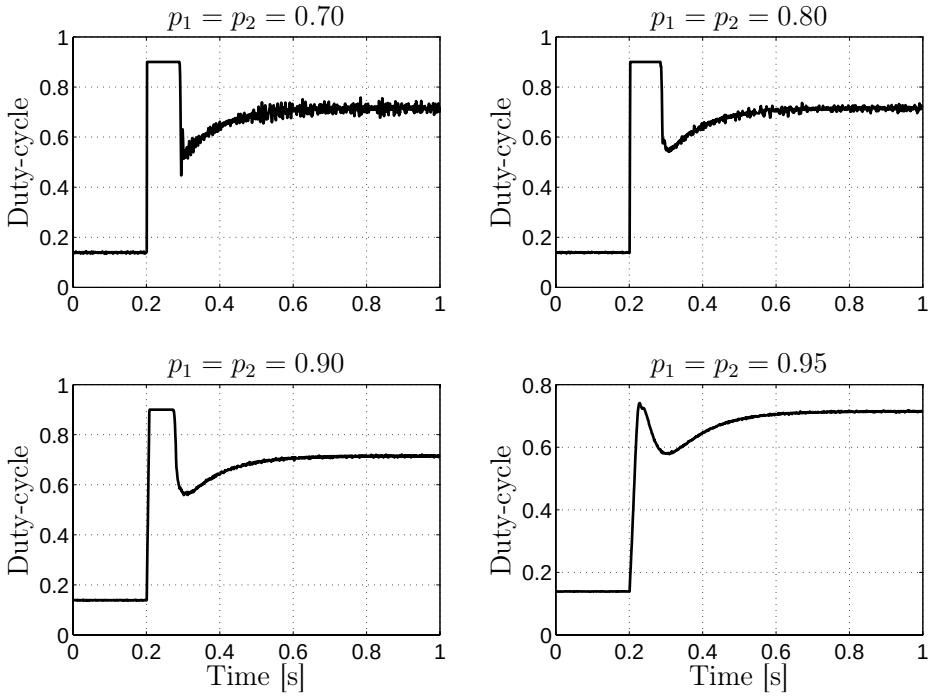


Figure 6.2: Corresponding system input to the output responses shown in Figure 6.1.

switching period, hence the duty-cycle may also change every fifth switching period and is kept constant in between.

The model parameter $b_{1,k}$ remains the same as in (5.21) except that (6.3) is used instead of (5.19) so that the steady-state behaviour (5.20) is still satisfied.

The measured (solid line) and desired (dashed line) output voltage responses are shown in Figure 6.3, whilst the corresponding control actions are shown in Figure 6.4. It is observed that at desired closed-loop pole locations of as slow as $p_1 = p_2 = 0.80$, input saturation is just not visible. Hence the range of physically achievable closed-loop pole locations is increased by increasing the controller sampling interval, however, it is to be noted that the maximal achievable slope of the output voltage cannot be affected since this is dependent on physical quantities determined by the circuit used, as discussed in Section 6.1.

Moreover, considering the control action for $p_1 = p_2 = 90$ shown in the lower left plot in Figure 6.4, a significant change in the behaviour is observable whilst the corresponding output voltage, lower left plot in Figure 6.3, accurately tracks the desired reference response, i.e. replicating the desired, linear closed-loop system output response. This observation might indicate the nonlinear behaviour of the system and that the SDP-PIP is able to handle the system nonlinearities satisfactorily.

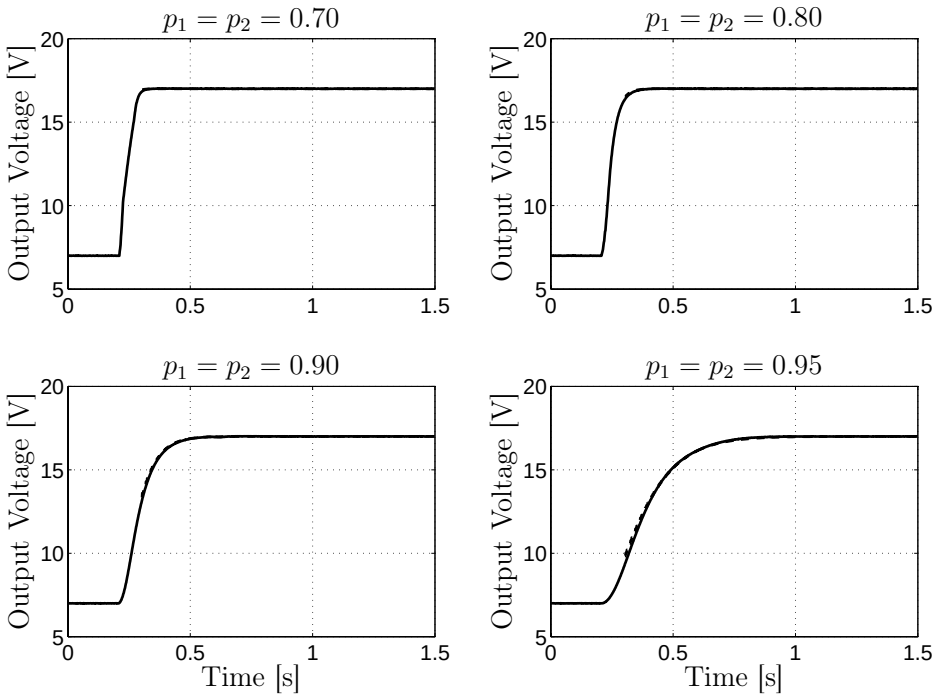


Figure 6.3: Desired output voltage response (dashed line) and measured output voltage response (solid line) to a step set-point change from 7 V to 17 V for various desired closed-loop pole locations and a SDP-PIP controller sampling interval of 5 ms.

Recall from the discussion in Section 5.2.3, that the model parameters $a_{1,k}$ and $b_{1,k}$, i.e. (6.3) and (5.21), respectively, are based on the previous output voltage measurement y_{k-1}. This is justified by the fact that the output voltage difference from one to the next sampling instance is sufficiently small. Increasing

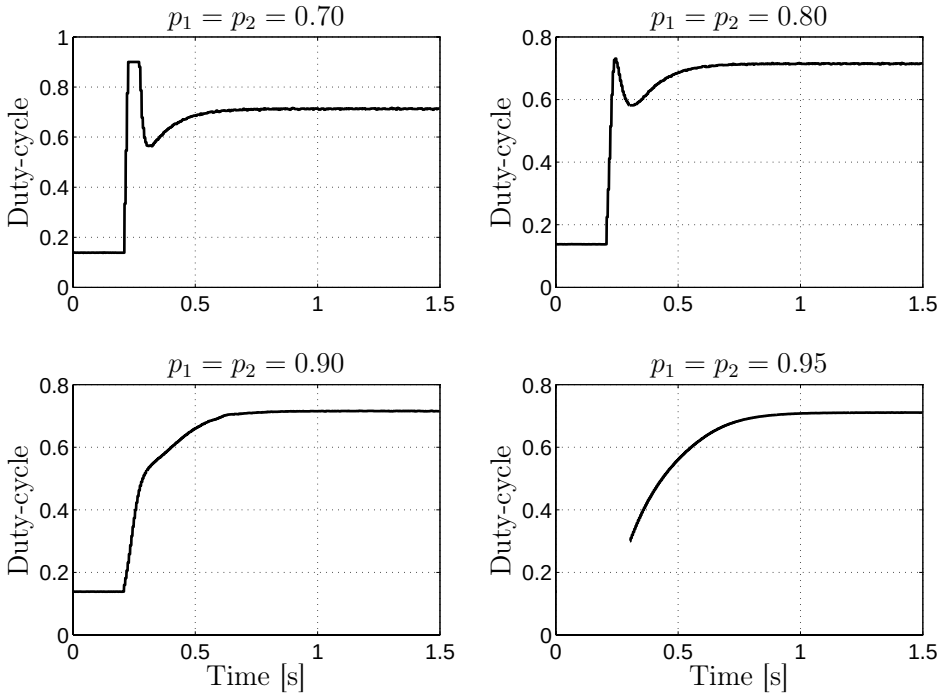


Figure 6.4: Corresponding system input to the output responses shown in Figure 6.3.

the controller sampling interval, however, may also increase the difference of the output voltage between consecutive sampling instances. Consequently, the controller performance may be impaired. By redefining the SDP model parameters (6.3) and (5.21) to be

$$\tilde{a}_{1,k} = -e^{\frac{-T_c}{\tau(y_{d,k},\, i_{R,k})}} \tag{6.4a}$$

and

$$\tilde{b}_{1,k} = \frac{y_{d,k}\ (1 + \tilde{a}_{1,k})}{y_{\infty}^{-1}(y_{d,k},\, i_{R,k})} \tag{6.4b}$$

the desired system output response, denoted $y_{d,k}$, which is given by

$$y_{d,k} = -d_1\ y_{d,k-1} - d_2\ y_{d,k-2} + (1 + d_1 + d_2)\ r_{k-1} \tag{6.5}$$

is incorporated into the SDP model parameters. Since the coefficients of the

desired characteristic equation d_1 and d_2 are determined by the a priori chosen closed-loop poles p_1 and p_2, hence known quantities, as well as the reference signal r_k $\forall k$, predictions of $y_{d,k}$ are available and can be used within the model parameters (6.4) and subsequently in the calculation of the SDP-PIP controller parameters (6.2).

Remark 6.3.1. *The use of $y_{d,k}$ in* (6.4) *is based on the assumption that the actual system output y_k is very close, if not identical, to the desired system output, i.e. $y_k \cong y_{d,k}$ $\forall k$.*

Quantified results, assessing the performance of the SDP-PIP controller when using the model parameters based on the measured system output, i.e. (6.3) and (5.21), respectively, as well as based on the desired system output (6.4), are presented in Tables 6.1 and 6.2, respectively. Additionally, the influence of the controller sampling interval T_c on the performance is evaluated for different desired closed-loop pole locations. The assessment criteria chosen are the mean of squared errors, denoted MSE, and similar to (5.25b), the mean integral of absolute errors, denoted IAE, which are given by

$$MSE = \frac{\|\boldsymbol{y} - \boldsymbol{y}_d\|_2^2}{N} \tag{6.6a}$$

and

$$IAE = \frac{|\boldsymbol{y} - \boldsymbol{y}_d|}{N} \tag{6.6b}$$

where $\boldsymbol{y} \in \mathbb{R}^{N\times 1}$ and $\boldsymbol{y}_d \in \mathbb{R}^{N\times 1}$ denote vectors of measured and desired system outputs, respectively.

Remark 6.3.2. *The main difference between the MSE and the IAE is the way they penalise the deviation of the measured output from the desired output. The IAE penalises this distance proportionally, while the MSE penalises this distance*

quadratically, which means that the greater the distance is the 'heavier' it is penalised compared to the IAE. On the other hand, for small deviations, the MSE does not penalise as 'heavy' as the IAE criterion.

Considering the case when the controller sampling interval is $T_c = 1\,\text{ms}$ and $p_1 = p_2 = 0.70,\ 0.80,\ 0.90$ as well as when $T_c = 5\,\text{ms}$ and $p_1 = p_2 = 0.70$, the system input saturates at its upper boundary, see Figures 6.2 and 6.4, the corresponding performance criteria quantities presented in Tables 6.1 and 6.2, both indicate that using the measured output in the model parameters, i.e. (6.3) and (5.21), is superior to using the desired output in the model parameters, i.e. (6.4), for calculating the controller parameters (6.2). This is as expected since, when the input is in saturation, the desired output is substantially different to the measured output, see also Figures 6.1 and 6.3, which results in model parameters being obtained, that do not correspond to the actual operating point of the system.

Now, consider the case when $p_1 = p_2 = 0.95$ and the system input does not saturate, i.e. the last row in Tables 6.1 and 6.2. Initially, focussing attention to the IAE criterion, i.e. Table 6.2, it can be observed that for $T_c = 1\,\text{ms}$ the values are almost identical when the measured and desired output is used. The performance discrepancy between using the measured and desired output, however, increases with increasing controller sampling interval T_c, whereby, using the desired output is superior. In Table 6.1, on the contrary, the greatest discrepancy is observable for $T_c = 10\,\text{ms}$, which is in accordance with the IAE criterion, while in the case of $T_c = 5\,\text{ms}$, almost identical MSE values for using the measured and desired output are obtained. This seems contradictory to the IAE results, however, it rather indicates that the discrepancies between the measured and desired system output are so small that they are almost not penalised by the MSE

but are more penalised by the IAE. Similarly in the case of $T_c = 1\,\text{ms}$, where the MSE values show a slightly greater discrepancy between using the measured and desired output than the IAE, where the values are almost identical.

Nevertheless, from the results, it can be said that when using a sufficiently small controller sampling interval, the performance of using the measured or desired, i.e. the one-step ahead predicted, system output value in the model parameters and subsequently for calculating the controller parameters, is very similar, whereby slight superiority on the side of using the desired system output is observable, which increases with increasing controller sampling interval. However, when the system input saturates, superiority is on the side of using the measured system output value.

In addition, note that the equivalent continuous-time pole locations differ from the discrete-time pole locations when different sampling intervals T_c are considered, although the numerical value of the discrete-time pole remains identical. However, since the performance criteria (6.6) evaluate the difference between the desired and respective measured closed-loop response, this accounts for the different transient behaviours at different sampling intervals T_c and numerical identical discrete-time closed-loop pole locations. Moreover, in Tables 6.1 and 6.2, the focus mainly lies on the performance evaluation when making use of the predicted, i.e. desired, system output value in the calculation of the controller parameters (6.2), i.e. by using (6.4), and making use of the measured system output for calculating the controller parameters (6.2), i.e. by using (6.3) and (5.21), when different controller sampling intervals are considered.

$p_1 = p_2$	$T_c = 1\,\text{ms}$		$T_c = 5\,\text{ms}$		$T_c = 10\,\text{ms}$	
	measured	desired	measured	desired	measured	desired
0.70	2.213	2.246	0.1667	0.2145	0.1544	0.0685
0.80	1.7303	1.755	0.0458	0.0143	0.0691	0.0262
0.90	0.7164	0.7804	0.0275	0.0274	0.0423	0.0386
0.95	0.0075	0.0039	0.0115	0.012	0.0411	0.0142

Table 6.1: Mean squared errors obtained for different desired closed-loop pole locations and different controller sampling intervals when the measured output voltage and the desired output voltage is used.

$p_1 = p_2$	$T_c = 1\,\text{ms}$		$T_c = 5\,\text{ms}$		$T_c = 10\,\text{ms}$	
	measured	desired	measured	desired	measured	desired
0.70	0.3955	0.3982	0.0934	0.0767	0.0886	0.0727
0.80	0.3505	0.353	0.0498	0.0138	0.0739	0.0571
0.90	0.2295	0.2418	0.0541	0.031	0.0795	0.0922
0.95	0.0223	0.0222	0.0501	0.0273	0.1152	0.0674

Table 6.2: Mean integral of absolute errors obtained for different desired closed-loop pole locations and different controller sampling intervals when the measured output voltage and the desired output voltage is used.

6.3.2 Load step regulation

Beside the dynamic behaviour of the output voltage for reference set-point changes, the behaviour of sudden load changes, i.e. output current changes, is considered next. This means that the output current drawn from the converter is a step from $i_R = 40\,\text{mA}$ to $i_R = 140\,\text{mA}$ and back, while the output voltage reference is kept at a constant value of $10\,\text{V}$.

SDP-PIP implementation results when using measured output voltage values in the model parameters and a controller sampling interval of $T_c = 5\,\text{ms}$ are shown in Figures 6.5 and 6.6. Making use of the reference, i.e. desired, output voltage in the model parameters, yields very similar results, as presented in Table 6.3. The fact that very slight differences between using the measured and the desired output voltage are observable is not surprising since the output

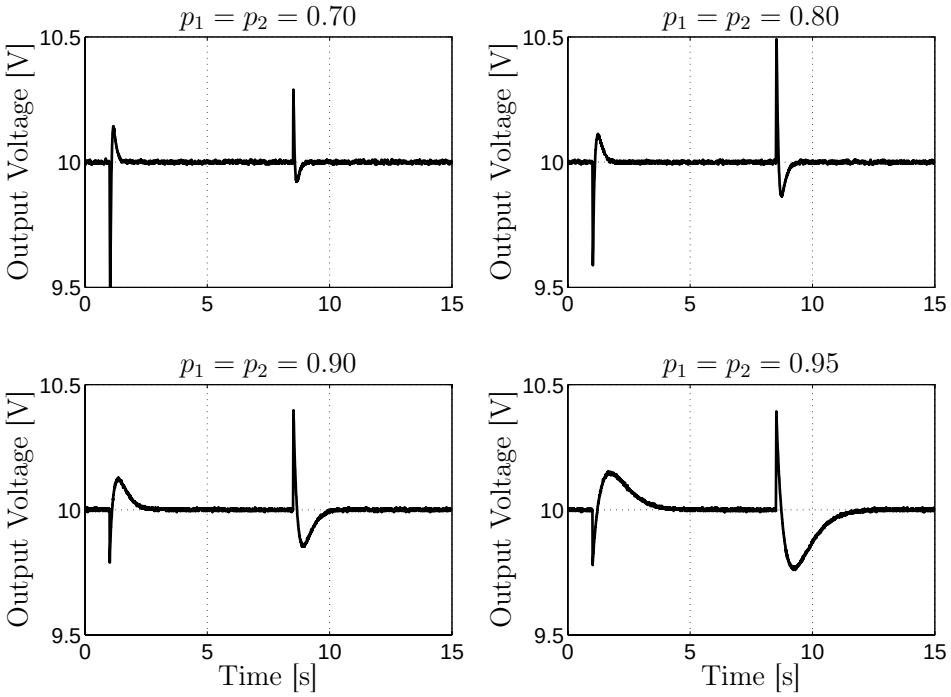


Figure 6.5: Output voltage responses to a output current step from $i_R = 40\,\text{mA}$ to 140 mA and back to 40 mA while the reference output voltage is kept constant at 10 V

voltage distortion, see Figure 6.5, caused by the load step, is sufficiently small, i.e. within $\pm 0.5\,\text{V}$, such that the impact on the model parameters and consequently on the controller gains is negligible. Moreover, identical results are not obtained due to the presence of measurement noise, which may also explain that the results obtained when using the, noise free, desired instead of the noisy, measured output voltage in the parameters are marginally superior.

Although the effect of the output voltage distortions on the control action can be viewed negligible, the system inputs are substantially different when the output current is at levels of $i_R = 40\,\text{mA}$ and $i_R = 140\,\text{mA}$, respectively, as it can be observed in Figure 6.6. This can be traced back to the model parameters (6.3), (5.21) and (6.4), which are also dependent on the output current and, consequently, determine the controller gains (6.2). On the contrary, it is

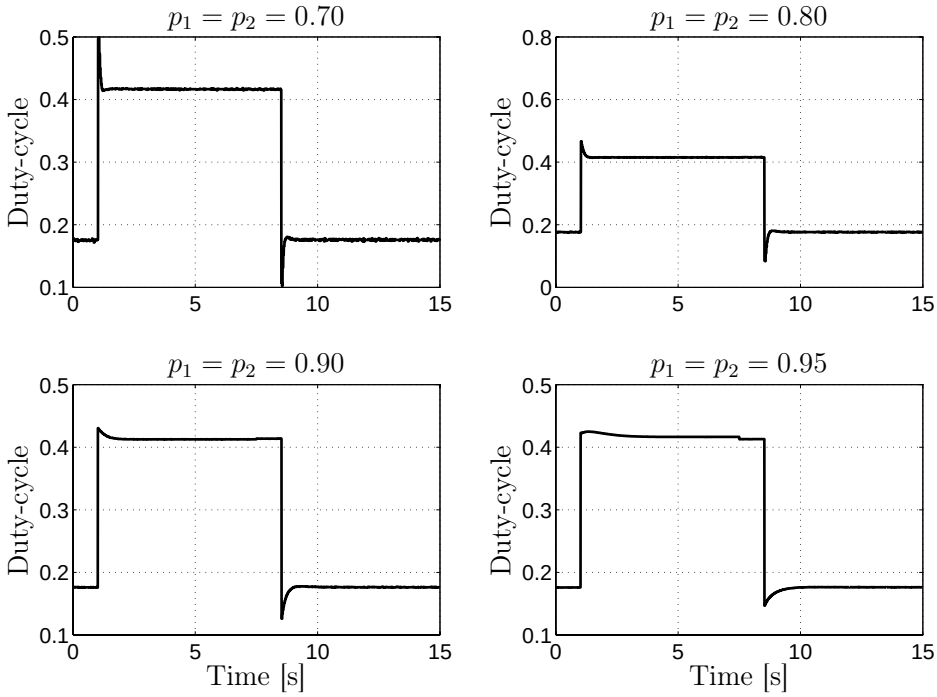


Figure 6.6: Corresponding system inputs to Figure 6.5

expected that a linear PIP is outperformed by the SDP-PIP due to the fixed controller gains, which do not allow this adaptation so that, consequently, mainly the integral action is required to drive the input to the respective levels. The integrator, however, integrates the error between the set-point and the measured output voltage, hence the output voltage distortion is necessary since the output current does not 'assist' as in the case of the SDP-PIP.

$p_1 = p_2$	$MSE \times 10^3$		$IAE \times 10^3$	
	measured	desired	measured	desired
0.70	0.9	0.1	7.4	4.1
0.80	1.3	1.1	9.9	9.3
0.90	1.6	1.6	15.6	14.3
0.95	4.5	5.1	35.5	38.7

Table 6.3: Quantified SDP-PIP implementation results when measured and desired output voltage is used in model parameters for load step at constant output reference voltage.

6.4 Comparison with the linear PIP controller

In this Section, the SDP-PIP implementation results are compared with the implementation results obtained by using linear PIP control. The controller gains of the linear PIP are fixed and calculated based on a linear model of the system obtained at a specific operating point. This linear model can be obtained by linearising the nonlinear SDP model at an appropriate operating point or, alternatively, using the 'frozen'/instantaneous linear model, which the SDP-PIP is using at the sampling instance corresponding to this specific operating point, cf. (Stables and Taylor 2006).

Initially, consider obtaining the linearised model. For this reason, however, an appropriate operating point of the system is required to be found at which the linearised model is based. This is chosen to be an operating point located in the centre, or close to it, of the operating range in order to keep the deviations from the chosen operating point throughout operation as small as possible.

The time constant of the linearised system is obtained from the equivalent time constants used in the SDP model shown in Figure 5.13. In order to obtain an appropriate value for the linearised model, the mean value of each trace shown in Figure 5.13 is calculated, yielding eleven values. From these mean values, again, the mean value is calculated, which is obtained to be $\tau_m = 0.0964\,\mathrm{s}$.

Recall that each trace in Figure 5.13 corresponds to a fixed output current value while covering the entire output voltage range, hence by obtaining the mean value of each trace, the mean value of the output voltage range corresponding to a certain output current is obtained. Moreover, since all the traces are, effectively, stepping through the output current operating range, the mean value of the mean values obtained for each individual trace can be considered as the mean value, i.e. aggregate, of the entire system operating range.

Mapping this time constant, i.e. τ_m, in discrete-time domain, the model parameter a_1 of the linearised model is given by

$$a_1 = -e^{\frac{-T_c}{\tau_m}} \tag{6.7}$$

The steady-state gain of the linearised model, in the following referred to as the process gain, is equal to the slope of the steady-state gain of the system (Stables and Taylor 2006).

The steady-state characteristic of the system is shown in Figure 5.10, where, also, each trace corresponds to a certain output current level. These individual steady-state characteristics are modelled by (5.10), which allows the associated process gains to be obtained by taking the derivatives of (5.10) w.r.t u_∞, i.e. $\{\frac{\partial}{\partial u_\infty}\, y_\infty(u_\infty)\}_i$ $i = 1, 2, \ldots, 11$, which denotes the sequence of process gains corresponding to the i output current levels. This is straightforward since (5.10) are cubic polynomials. Subsequently, in the same manner as above, the mean value of the mean values, i.e. aggregate, of the derivatives of each trace is obtained. Consequently, by taking (5.20) into account, the process gain of the linearised system model is given by

$$\frac{1}{N}\sum_{i=1}^{N} E\left[\left\{\frac{\partial y_\infty}{\partial u_\infty}\right\}_i\right] = 27.421 = \frac{b_1}{1+a_1} \qquad \text{with} \quad N = 11 \tag{6.8}$$

where $E[\,\cdot\,]$ denotes the expected value, i.e. mean value, and subsequently the model parameter b_1 is obtained as

$$b_1 = 27.421(1 + a_1) \tag{6.9}$$

Both the instantaneous linear model and linearised model share the model para-

meter a_1 (6.7). These models differ, however, in the remaining model parameter b_1. The parameter b_1 of the instantaneous linear model, in the following denoted $\hat{b}_1$, is associated with the steady-state gain of the SDP model and obtained by, initially, calculating the mean value of the mean values of the steady-state gains corresponding to each trace shown in Figure 5.10 and, subsequently, by taking (5.20) into account, similar to (6.8),

$$\frac{1}{N}\sum_{i=1}^{N} E\left[\left\{\frac{y_\infty}{u_\infty}\right\}_i\right] = 46.997 = \frac{\hat{b}_1}{1+a_1} \qquad \text{with} \quad N = 11 \tag{6.10}$$

consequently, the parameter $\hat{b}_1$ of the instantaneous linear model is obtained as

$$\hat{b}_1 = 46.997(1 + a_1) \tag{6.11}$$

It is to be noted that the steady-state gain cannot be obtained directly from Figure 5.10 since the steady-state gain is given by the fraction $\frac{y_\infty}{u_\infty}$, hence the traces observable in Figure 5.10, which show the steady-state output as a function of the steady-state input (5.10), are required to be divided by their respective steady-state input, cf. (6.10), in order to obtain the steady-state gain.

Remark 6.4.1. *The use of the aggregated median values in* (6.7)*,* (6.9) *and* (6.11)*, respectively, instead of the mean values, could equally be justified.*

Implementation results of the linear PIP controller based on the linearised model, i.e. (6.7) and (6.9), as well as the linear PIP based on the instantaneous linear model, i.e. (6.7) and (6.11), compared with the SDP-PIP controller, is shown in Figures 6.7 and 6.8, respectively. The controller sampling interval is chosen to be $T_c = 5\,\text{ms}$ so that, effectively, the SDP-PIP controller implementation, as shown in Figures 6.3 and 6.4, respectively, is compared with the linear PIP controllers. Quantified performance results of the linear PIP controller per-

formances are presented in Table 6.4, whereby the performance of the SDP-PIP can be obtained from Tables 6.1 and 6.2, respectively.

In Figure 6.7, it can be observed that both linear PIP controllers achieve very similar performance results as the SDP-PIP when a reasonably fast closed-loop poles location is chosen, e.g. $p_1 = p_2 = 0.80$. The slower these closed-loop poles are chosen, i.e. $p_1 = p_2 = 0.90,\ 0.95$, the less accurate the linear PIP controllers are tracking the desired response. The SDP-PIP controller, on the contrary, demonstrates its accuracy throughout a wider range of operation.

Moreover, from the quantified results presented in Table 6.4, it can be said that the linear PIP controller based on the linearised model outperforms the linear PIP based on the instantaneous linear model, upon which the SDP-PIP makes use, as long as the input does not saturate. Input saturation can be observed in Figure 6.8 for $p_1 = p_2 = 0.70$. Although the linearised model achieves a superior performance throughout the operating range than the instantaneous linear model, both are not able to outperform the SDP-PIP, which, interestingly has more in common with the instantaneous linear model based PIP than with PIP based on the linearised model. The PIP based on the instantaneous linear model, however, is able to achieve satisfactory performance at a certain 'operating point', see Figure 6.7 when $p_1 = p_2 = 0.80$. This may explain the superiority of the SDP-PIP, where controller gains are calculated based on the instantaneous linear model at each sampling instance. On the other hand, since the performance of the PIP based on the instantaneous linear model rapidly decreases when deviating from this 'operating point', this then also applies to the SDP-PIP in the presence of model parameter uncertainties and, as a consequence, may result in increased sensitivity to model parameter uncertainties, as observed in Section 4.3.

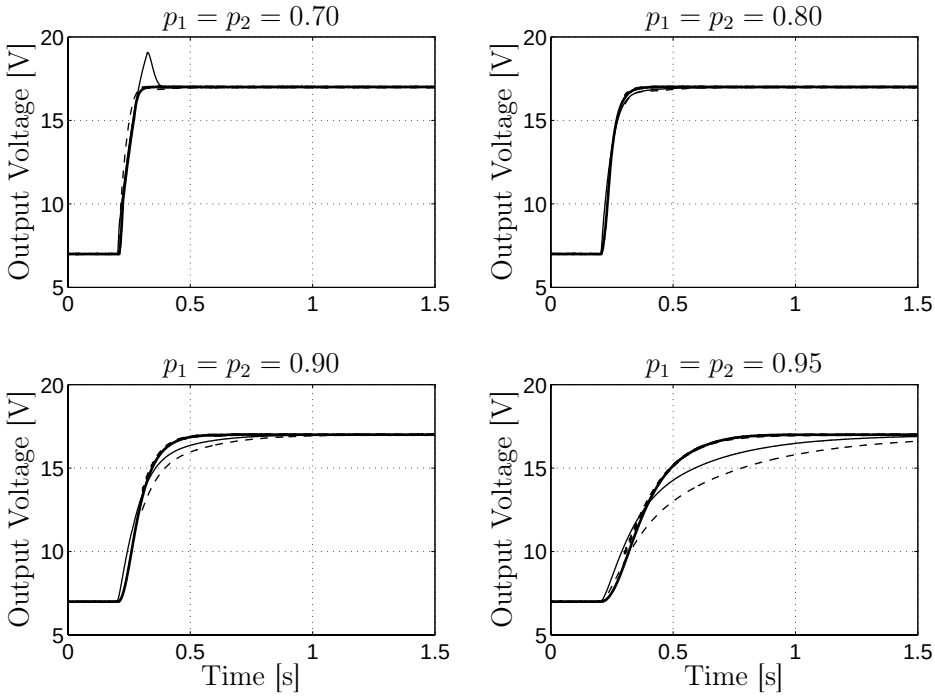


Figure 6.7: Desired output voltage response (thick dashed line) and measured output voltages using the SDP-PIP controller (thick solid line), linear PIP controller based on the linearised model (thin solid line) and linear PIP controller based on the instantaneous linear model (thin dashed line).

6.4.1 Comparison load step regulation

The load step, i.e. output current step, distortion regulation of the SDP-PIP, as shown in Figures 6.5 and 6.6, respectively, is compared against the linear PIP controllers introduced above.

Implementation results of the linear PIP controllers compared with the SDP-PIP controller are shown in Figures 6.9 and 6.10, respectively. It can be observed that the SDP-PIP controller clearly outperforms the linear PIP controllers, which is as expected in the discussion in Section 6.3.2. This observation is also confirmed when comparing the quantified performance results of the linear PIP controllers, presented in Table 6.5, with those of the SDP-PIP controller, presented in Table 6.3.

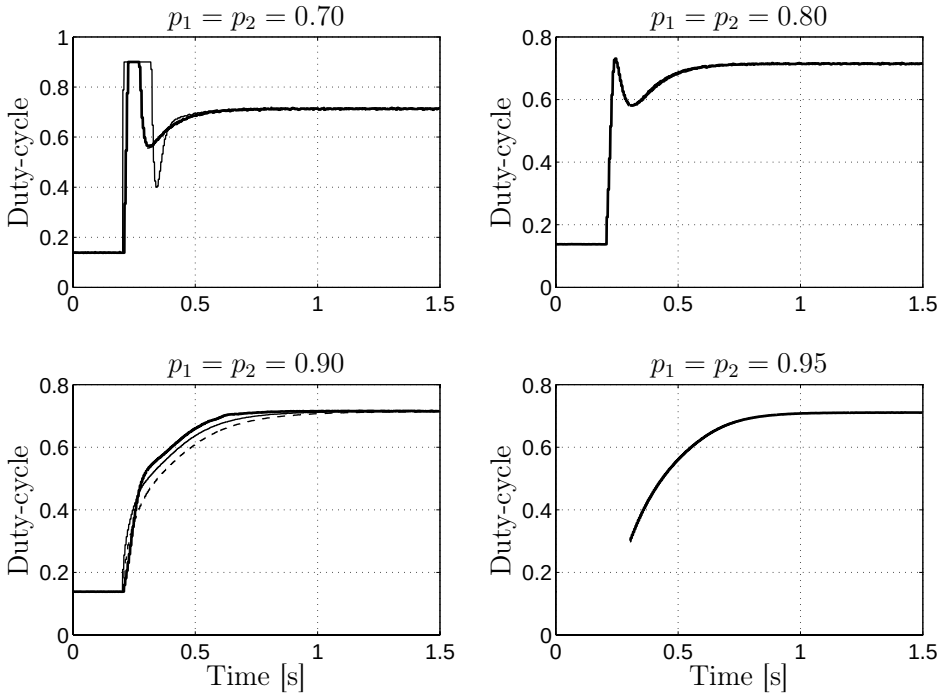


Figure 6.8: System inputs, i.e. control actions, corresponding to the output responses shown in Figure 6.7

Furthermore, the output voltage responses shown in Figure 6.7 indicate a slower response of the linear PIP controllers compared to the SDP-PIP controller. Hence it is not surprising that the linear PIP controllers also regulate the output voltage distortion slower than the SDP-PIP controller. Considering the case of $p_1 = p_2 = 0.80$, in Figure 6.7, very similar performance of the linear PIP controllers and SDP-PIP controller is observable, while the SDP-PIP controller clearly outperforms the linear PIP controllers in the case of load step regulation, see upper right plot in Figure 6.9 (also c.f. Figure 6.5). This confirms the anticipation that the adaptation of the SDP-PIP controller gains, caused by the output current dependency, significantly improves the performance of the SDP-PIP controller compared to the linear PIP controller.

$p_1 = p_2$	Linearised model		Instantaneous linear model	
	MSE	IAE	MSE	IAE
0.70	0.248	0.1356	0.0434	0.0352
0.80	0.0154	0.0559	0.0394	0.0705
0.90	0.0623	0.1583	0.2563	0.2831
0.95	0.3375	0.4617	1.6087	1.0053

Table 6.4: Quantified linear PIP implementation results for comparison with SDP-PIP as shown in Figures 6.7 and 6.8

$p_1 = p_2$	Linearised model		Instantaneous linear model	
	$MSE \times 10^3$	$IAE \times 10^3$	$MSE \times 10^3$	$IAE \times 10^3$
0.70	4.5	15.7	11	23.2
0.80	13.1	29.6	33.6	48.9
0.90	98.5	111.1	245.2	191
0.95	800.4	441.7	1883	782.3

Table 6.5: Quantified linear PIP implementation results of load step regulation for comparison with the SDP-PIP as shown in Figure 6.5

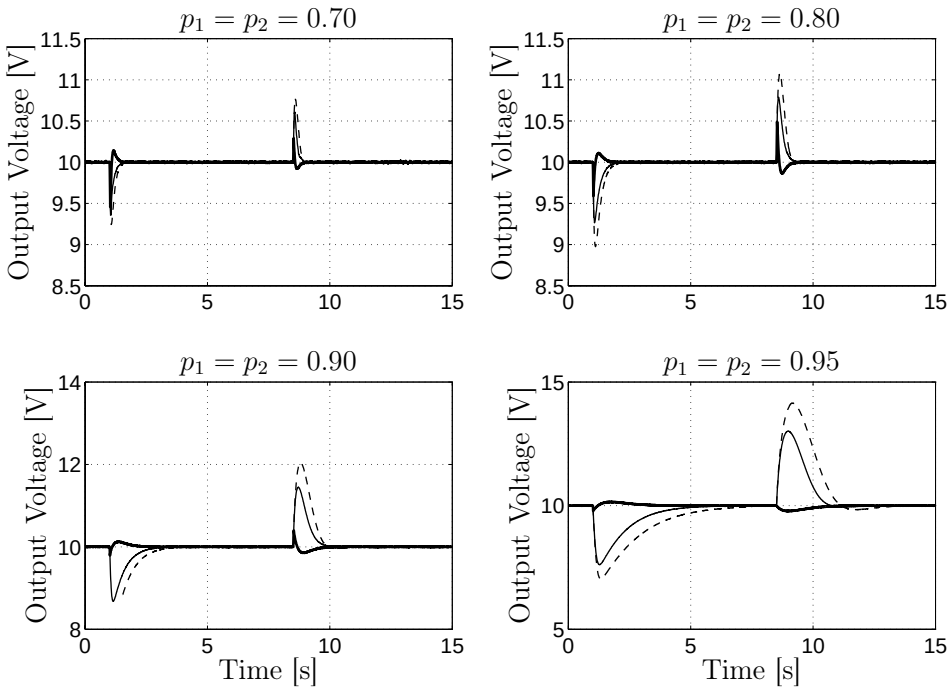


Figure 6.9: Measured output voltage responses using the SDP-PIP controller (thick solid line), linear PIP controller based on the linearised model (thin solid line) and linear PIP controller based on the instantaneous linear model (thin dashed line) to an output current step from $i_R = 40\,\text{mA}$ to $140\,\text{mA}$ and back to $40\,\text{mA}$

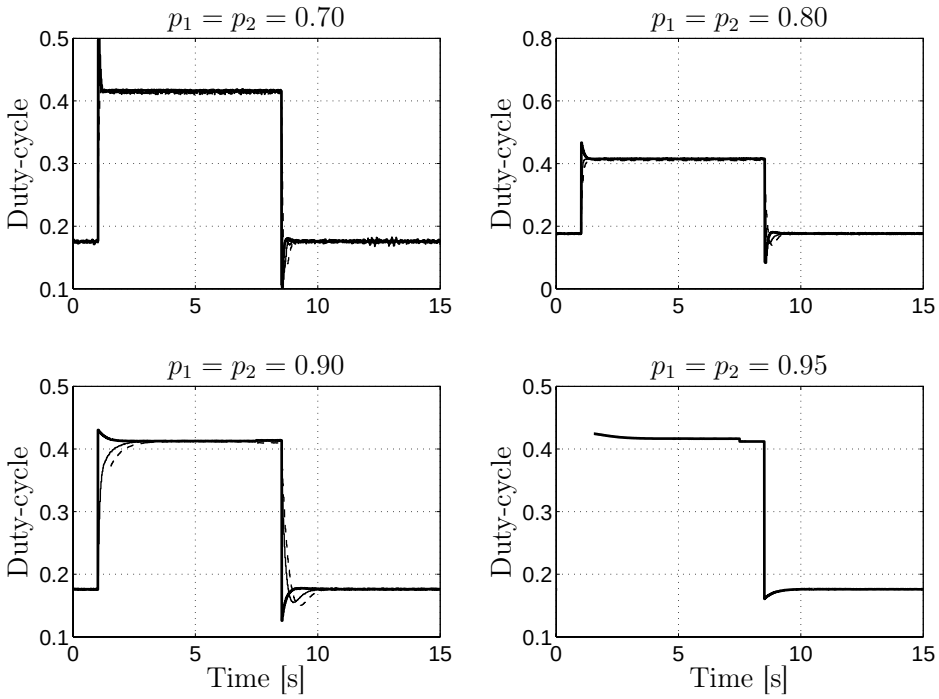


Figure 6.10: System inputs, i.e. control actions, corresponding to the output responses shown in Figure 6.9

6.5 Concluding remarks

Based on the SDP model for the DC-DC boost converter elaborated in the previous Chapter, in this Chapter, a model-based output voltage control strategy has been proposed making use of the SDP-PIP controller structure. The output voltage dynamic behaviour for voltage reference set-point changes and, caused by sudden output current, i.e. load changes, output voltage distortion regulation have been considered.

Also, the performance of the SDP-PIP control approach has been compared with the performance of linear PIP controllers. The gains of these linear controllers are obtained based on a linearised model and on an instantaneous linear model, which can be obtained from the SDP model directly.

The steady-state gain of the linearised model is based on the slope of the system steady-state gain while the instantaneous linear model uses the steady-state gain of the system at the chosen operating point directly.

The operating point at which these linear models are based is chosen such that a centrally located operating point in the operating range is obtained.

Furthermore, the superiority and efficacy of the SDP-PIP control approach over the linear PIP controllers is experimentally demonstrated.

Chapter 7

Conclusions and Further Work

Conclusions of the work carried out and documented in this Thesis are given in Section 7.1. In summary, this concerns the proposed methodological approach to model-based state dependent parameter (SDP) control, in particular SDP proportional-integral-plus (PIP) pole-assignment control, the proposed decoupling strategy using linear model-based predictive control (MPC) based on MIMO non-minimal state-space (NMSS) models, a SDP modelling approach for a DC-DC boost converter operating in discontinuous conduction mode (DCM), which is compared to a Hammerstein-bilinear structured (HBS) modelling approach and finally, the experimental demonstration of SDP-PIP pole-assignment control to the DC-DC boost converter based on the developed SDP model.

Since research is an ongoing, if not a never ending process, the research carried out in this Thesis has taken steps further but, nevertheless, as with all research never completed to the final extent, hence in the authors view, potentially fruitful directions for further research are suggested in Section 7.2.

7.1 Conclusions

This Section provides a comprehensive summary of the key achievements that have arisen in this Thesis in the order of their importance.

7.1.1 SDP discrete-time pole-assignment control

In Chapter 4 of this Thesis, further developments in nonlinear model based control have been achieved, where the focus has been on a class of discrete-time SDP system models in conjunction with the SDP-PIP control methodology using pole-assignment (Taylor et al. 2009). Since pole-assignment, being a linear control technique, applied to nonlinear SDP modelled systems is considered, it is paradoxically clear that this technique may only be applied to a linear model, hence to the time-step instantaneous linear, i.e. 'frozen' in time, system model. This means, that the nonlinear model is to be considered linear at every sampling instance, while nonlinear overall, i.e. in its evolution. This issue has been addressed in this Thesis. An approach has been proposed where an additional conceptual time shift operator is introduced accounting for shifting the entire nonlinear system in time while, at each sampling instance, the 'standard' time shift operator only affects the instantaneous linear model. This approach provides clarity for the paradoxical issue of applying linear pole-assignment to a nonlinear system model. It has been shown that by making use of this conceptual time shift operator, the relationship between linear and SDP-PIP can be explained, as well as the derivation of the SDP-PIP control law, also using the incremental input formulation. Additionally, the results obtained also coincide with the results in (Taylor et al. 2009), hence the proposed approach can be viewed as an extension which adds clarity to the body of work on this topic, e.g. as presented in (Taylor et al. 2009).

Furthermore, the clarity of interpretation afforded by the conceptual time shift operator, led to the development of a strategy for generalising discrete-time SDP-PIP pole-assignment control, where the SDP model exhibits equivalent system zeros, i.e. the instantaneous linear model contains system zeros. The approach uses cancellation of the closed-loop system zeros of the instantaneous linear system model at each sampling instance. However, it has been shown that in the case of model parameter uncertainties, i.e. mismatch, the performance is significantly impaired, as observed for one of the simulation examples studied in this Thesis. In addition, it has been pointed out that the robustness also depends on a number of other factors, including the desired closed-loop poles, set-point sequence and the nature of the SDPs.

7.1.2 MIMO decoupling NMSS MPC control

In Chapter 3 of this Thesis, a MPC decoupling control strategy for linear, square MIMO systems has been proposed, where the system model on which the MPC is based, is of a NMSS form. Moreover, the NMSS MPC in incremental input form (Wang and Young 2006) and the form of using an integral-of-errors state variable (Exadaktylos et al. 2006) has been considered. Apart from NMSS MPC system output decoupling approaches based on weighting matrix optimisation (Exadaktylos and Taylor 2010), in this Thesis, an analytic decoupling method, which achieves its goal by closed-loop system model diagonalisation via an input transformation, adopted from (Plummer and Vaughan 1997; Kubalcik and Bobal 2006), has been transferred into the NMSS MPC framework.

Furthermore, a modification to the formulation of the NMSS MPC in incremental input form has been proposed, which allows one to consider individually the transformed system input - system output pairs so that individual

control and prediction horizons can be assigned to these pairs. These horizons can be viewed as additional tuning parameters, hence provides a greater flexibility. However, when applied to the integral-of-errors state variable form, due to the implicit consideration of the future set-point trajectory in the respective MPC cost function, this modification cannot be straightforwardly applied.

Additionally, it has been shown that the ability of considering the input-output pairs individually, allows imposed constraints to be handled without inducing cross-coupling effects in the system outputs. It has been proposed to make use of the reference trajectory adaptation method (Bemporad and Mosca 1994), which, in combination with the modified NMSS MPC in incremental input form, effectively provides freedom to choose which system output is allowed to account for the imposed constraints. The integral-of-errors state variable formulation, however, is not able to provide this flexibility.

Finally, by making use of a simulation example, the impact of model parameter uncertainties has been evaluated, where it could be shown that the integral-of-errors state variable form and incremental input form achieve very similar performance. However, due to the analytic decoupling method, model parameter uncertainties may cause cross-coupling effects to re-emerge.

7.1.3 SDP modelling and control of a DC-DC boost converter operating in DCM

As an application example, a SDP modelling approach for a DC-DC boost converter operating in DCM has been developed in Chapter 5, upon which SDP-PIP control has subsequently been implemented in Chapter 6. This modelling approach is based on measured input-output data rather than being deduced from physical relationships. The required data for modelling were acquired from

laboratory based experiments, as well as for the model verification. The proposed SDP parameter identification procedure is based on acquired data while stepping through the entire pre-defined operating range and extracting the steady-state as well as dynamic behaviour of the converter. Additionally, there exists not only a relationship between the system input and system output, i.e. duty-cycle and output voltage, respectively, but also the converter behaviour is influenced by the output current drawn, which has been taken into account by considering the input-output relation at different output current levels throughout the output current range and subsequent interpolation by making use of polynomials.

The proposed SDP model has been compared with a HBS modelling approach, which accounts for the steady-state behaviour in the Hammerstein static nonlinear term and the dynamics in the bilinear term, which is conceptually similar to the SDP approach. In order to account for the influence of the output current, a sequence of HBS sub-models has been identified at different output current levels and these sub-models are subsequently blended by making use of Gaussian membership functions so that an 'overall' HBS model is obtained.

Although these modelling approaches are similar, it has been shown that the proposed SDP modelling approach achieves slightly superior performance results. Moreover, due to the Gaussian blending of the sub-models, the SDP model could be viewed as structurally more beneficial regarding subsequent model-based control since the 'interpolation' is handled inherently in the SDP model parameters and not comprised of several sub-models.

Furthermore, the developed SDP model has been used as a basis for subsequent SDP-PIP pole-assignment control implementation and the efficacy has been experimentally demonstrated. In addition, the SDP-PIP performance has been compared to the performance obtained by making use of linear PIP

pole-assignment control. A linear PIP controller based on the instantaneous linear model, i.e. the time-step 'frozen' model upon which the SDP-PIP is based at each sampling instance, as well as a linear PIP controller based on a linearised model, have been used. The operating point assumed for the instantaneous linear and linearised model is chosen such that it lies centrally inside the operating range. Moreover, it has been demonstrated, not surprisingly, that the SDP-PIP clearly outperforms both linear PIP controllers when the entire operating range is considered. This observation applies to regulation of load steps as well as output voltage set-point changes.

Additionally, it has been observed that the PIP controller based on the linearised model is slightly superior to that based on the instantaneous linear model and, finally, the physical performance constraints of the converter have been highlighted.

7.2 Further work

In this Section, some directions for further research are suggested.

7.2.1 SDP model based control

The SDP model based control approaches, e.g. SDP-PIP, SDRE, etc. utilise linear control techniques performed at each sampling instance, hence the focus is on the instantaneous linear model. Consequently, attention to the SDP system behaviour between the consecutive sampling instances may be required in order to explore and explain the evolution of the, in fact, nonlinear system. Hence the interaction between these levels of consideration, i.e. the instantaneous linear and overall nonlinear level, constitutes an open research question.

Furthermore, regarding the SDP-PIP, the issue of dealing with equivalent system numerator zeros is not solved to the last extent. There is still a lack of robustness concerning model parameter uncertainties as well as dealing with SDP modelled systems that exhibit non-minimum phase behaviour.

Although developments regarding the SDP MPC have been proposed, e.g. (Exadaktylos 2007, Chap. 8), research on this topic is still at an early stage.

7.2.2 Decoupling NMSS MPC

It has been pointed out that the constraints imposed on the NMSS-MPC form using an integral-of-errors state variable induce system output cross-coupling effects, hence further research on the constraints formulation such that these cross-coupling effects are suppressed might be useful. Moreover, the investigation of the effects of disturbances and their impact on the decoupling strategy has been left as an open question.

In addition, explicit model parameter uncertainty formulations could be incorporated into the decoupling strategy such that the impact of parameter uncertainties is reduced.

Also, so far, linear NMSS MPC has been considered so that research on transferring this approach into a nonlinear framework, i.e. SDP MPC, might be interesting.

7.2.3 SDP model based control of a DC-DC boost converter

Regarding the SDP-PIP control of a DC-DC boost converter operating in DCM, further considerations and consequently incorporation of constraints into the control strategy might lead to further improvements. Also, considering the dynamics,

introduced by the output current, in the SDP model is a potential improvement.

Furthermore, the SDP modelling and control approach can be extended to the continuous conduction operation mode (CCM). The boost converter operating in CCM, however, exhibits a non-minimum phase behaviour which is, at the current stage of research, an obstacle concerning SDP-PIP pole-assignment control, which is required to be overcome first. However, investigations on the applicability of alternative SDP control methods such as the SDRE approach may be of interest.

Finally, an investigation of applying SDP modelling and control methods to different DC-DC switched mode converter topologies, e.g. Buck, Buck-Boost, SEPIC etc., remains an open research question.

Appendix A

A.1 Computation of SDP-PIP closed loop parameters γ and β

In the following, the computation of the parameters $\gamma_{i,k}$ and $\beta_{j,k}$ in the closed-loop equation (4.27), as presented in (4.28), is shown.

Recall from (4.25) that the closed loop system is given by

$$\begin{aligned} A_{k+\tau}z^{\tau}y_k + b_{\tau,k+\tau}F_k y_k + b_{\tau,k+\tau}\frac{K_{I,k}}{\Delta}y_k = b_{\tau,k+\tau}\frac{K_{I,k}}{\Delta}r_k \\ + (b_{\tau+1,k+\tau} - b_{\tau,k+\tau}g_{1,k})u_{k-1} + (b_{\tau+2,k+\tau} - b_{\tau,k+\tau}g_{2,k})u_{k-2} \\ + \ldots + (b_{\tau+m,k+\tau} - b_{\tau,k+\tau}g_{m,k})u_{k-m} - g_{m+1,k}b_{\tau,k+\tau}u_{k-m-1} \\ - \ldots - g_{m+\tau-1,k}b_{\tau,k+\tau}u_{k-m-\tau+1} \end{aligned} \tag{A.1}$$

Also, recall from (4.28) that the computation is an iterative procedure, i.e. (A.1) is iteration zero. So, (A.1) can be reformulated to be

$$\begin{aligned} A_{k+\tau}z^{\tau}y_k + b_{\tau,k+\tau}F_k y_k + b_{\tau,k+\tau}\frac{K_{I,k}}{\Delta}y_k - b_{\tau,k+\tau}\frac{K_{I,k}}{\Delta}r_k \\ = c_1^{(0)}u_{k-1} + c_2^{(0)}u_{k-2} + \ldots + c_{m+\tau-1}^{(0)}u_{k-m-\tau+1} \end{aligned} \tag{A.2}$$

with $c_i^{(0)} = b_{\tau+i,k+\tau} - b_{\tau,k+\tau}g_{i,k}$ $i = 1, 2, \ldots, m+\tau-1$ and $b_{\tau+i,k+\tau} = 0\ \forall i > m$. The

superscript $^{(\cdot)}$ denotes the iteration. Now, sequentially substituting the previous system inputs $u_{k-i}\ i = 1, 2, \ldots, \tau-1$ obtained from the open-loop system equation (4.1) and under consideration of Example 4.1, given by

$$\begin{aligned}
u_{k-1} &= \frac{1}{b_{\tau,k+\tau-1}} A_{k+\tau-1}\ z^{\tau-1} y_k - \frac{b_{\tau+1,k+\tau-1}}{b_{\tau,k+\tau-1}} u_{k-2} - \frac{b_{\tau+2,k+\tau-1}}{b_{\tau,k+\tau-1}} u_{k-3} \\
&\quad - \ \ldots \ - \frac{b_{\tau+m,k+\tau-1}}{b_{\tau,k+\tau-1}} u_{k-m-1} \\
u_{k-2} &= \frac{1}{b_{\tau,k+\tau-2}} A_{k+\tau-2}\ z^{\tau-2} y_k - \frac{b_{\tau+1,k+\tau-2}}{b_{\tau,k+\tau-2}} u_{k-3} - \frac{b_{\tau+2,k+\tau-2}}{b_{\tau,k+\tau-2}} u_{k-4} \\
&\quad - \ \ldots \ - \frac{b_{\tau+m,k+\tau-2}}{b_{\tau,k+\tau-2}} u_{k-m-2} \\
&\ \ \vdots \\
u_{k-\tau+1} &= \frac{1}{b_{\tau,k+1}} A_{k+1}\ z\ y_k - \frac{b_{\tau+1,k+1}}{b_{\tau,k+1}} u_{k-\tau} - \frac{b_{\tau+2,k+1}}{b_{\tau,k+1}} u_{k-\tau-1} \\
&\quad - \ \ldots \ - \frac{b_{\tau+m,k+1}}{b_{\tau,k+1}} u_{k-m-\tau+1}
\end{aligned} \tag{A.3}$$

so that substituting u_{k-1} from (A.3) into (A.2) and re-arranging, yields

$$\begin{aligned}
\ldots = &\frac{c_1^{(0)}}{b_{\tau,k+\tau-1}} A_{k+\tau-1} z^{\tau-1} y_k + \left(c_2^{(0)} - c_1^{(0)} \frac{b_{\tau+1,k+\tau-1}}{b_{\tau,k+\tau-1}} \right) u_{k-2} \\
&+ \left(c_3^{(0)} - c_1^{(0)} \frac{b_{\tau+2,k+\tau-1}}{b_{\tau,k+\tau-1}} \right) u_{k-3} + \ldots + \left(c_{m+1}^{(0)} - c_1^{(0)} \frac{b_{\tau+m,k+\tau-1}}{b_{\tau,k+\tau-1}} \right) u_{k-m-1} \\
&+ c_{m+2}^{(0)} u_{k-m-2} + \ldots + c_{m+\tau-1}^{(0)} u_{k-m-\tau+1}
\end{aligned} \tag{A.4}$$

where, here, and in the following, for the sake of brevity, the left-hand side of (A.2) is dropped, which, however, remains unchanged. Since (A.4) can be seen as the first iteration, similarly to (A.2), it can be reformulated as

$$\ldots = \gamma_{1,k} A_{k+\tau-1} z^{\tau-1} y_k + c_2^{(1)} u_{k-2} + \ldots + c_{m+\tau-1}^{(1)} u_{k-m-\tau+1} \tag{A.5}$$

where $c_i^{(1)} = c_i^{(0)} - c_1^{(0)} \dfrac{b_{\tau+i-1,k+\tau-1}}{b_{\tau,k+\tau-1}}\ \forall i = 2, 3, \ldots, m+1$ and since $b_{\tau+m+j,k+\tau-1} =$

$0\ \forall j > 0$, note that $c^{(1)}_{m+i} = c^{(0)}_{m+i}\ \forall i = 2, 3, \ldots, \tau - 1$. Also, the closed-loop parameter $\gamma_{1,k}$ is given by $\gamma_{1,k} = \dfrac{c^{(0)}_1}{b_{\tau,k+\tau-1}}$.

Then, in the next iteration, substituting u_{k-2} from (A.3) into (A.5) and re-arranging, gives

$$\begin{aligned}\ldots =&\gamma_{1,k}A_{k+\tau-1}z^{\tau-1}y_k + \frac{c^{(1)}_2}{b_{\tau,k+\tau-2}}A_{k+\tau-2}z^{\tau-2}y_k + \left(c^{(1)}_3 - c^{(1)}_2\frac{b_{\tau+1,k+\tau-2}}{b_{\tau,k+\tau-2}}\right)u_{k-3}\\ &+ \left(c^{(1)}_4 - c^{(1)}_2\frac{b_{\tau+2,k+\tau-2}}{b_{\tau,k+\tau-2}}\right)u_{k-4} + \ldots + \left(c^{(1)}_{m+2} - c^{(1)}_2\frac{b_{\tau+m,k+\tau-2}}{b_{\tau,k+\tau-2}}\right)u_{k-m-2}\\ &+ c^{(1)}_{m+3}u_{k-m-3} + \ldots + c^{(1)}_{m+\tau-1}u_{k-m-\tau+1}\end{aligned} \tag{A.6}$$

which becomes, similarly to (A.5),

$$\begin{aligned}\ldots = \gamma_{1,k}A_{k+\tau-1}z^{\tau-1}y_k + \gamma_{2,k}A_{k+\tau-2}z^{\tau-2}y_k&\\ + c^{(2)}_3u_{k-3} + c^{(2)}_4u_{k-4} + \ldots + c^{(2)}_{m+\tau-1}u_{k-m-\tau+1}&\end{aligned} \tag{A.7}$$

where $\gamma_{2,k} = \dfrac{c^{(1)}_2}{b_{\tau,k+\tau-2}}$, $c^{(2)}_i = c^{(1)}_i - c^{(1)}_2\dfrac{b_{\tau+i-2,k+\tau-2}}{b_{\tau,k+\tau-2}}\ \forall i = 3, 4, \ldots, m+2$ and $c^{(2)}_{m+i} = c^{(1)}_{m+i} = c^{(0)}_{m+i}\ \forall i = 3, 4, \ldots, \tau - 1$.

Proceeding in this manner, finally, at iteration $\tau - 1$, i.e. substituting $u_{k-\tau+1}$, yields

$$\begin{aligned}\ldots =&\gamma_{1,k}A_{k+\tau-1}z^{\tau-1}y_k + \gamma_{2,k}A_{k+\tau-2}z^{\tau-2}y_k + \ldots + \gamma_{\tau-2,k}A_{k+2}z^2y_k\\ &+\frac{c^{(\tau-2)}_{\tau-1}}{b_{\tau,k+1}}A_{k+1}z^1y_k + \left(c^{(\tau-2)}_{\tau} - c^{(\tau-2)}_{\tau-1}\frac{b_{\tau+1,k+1}}{b_{\tau,k+1}}\right)u_{k-\tau}\\ &+ \left(c^{(\tau-2)}_{\tau+1} - c^{(\tau-2)}_{\tau-1}\frac{b_{\tau+2,k+1}}{b_{\tau,k+1}}\right)u_{k-\tau-1} + \ldots\\ &+ \left(c^{(\tau-2)}_{\tau+m-1} - c^{(\tau-2)}_{\tau-1}\frac{b_{\tau+m,k+1}}{b_{\tau,k+1}}\right)u_{k-\tau-m+1}\end{aligned} \tag{A.8}$$

which, as well, can be written as

$$\ldots = \sum_{i=1}^{\tau-1} \gamma_{i,k} A_{k+\tau-i} z^{\tau-i} y_k + \sum_{i=\tau}^{\tau+m-1} \Bigg(c_i^{(\tau-1)} u_{k-i} \tag{A.9}$$

where $\gamma_{\tau-1,k} = \dfrac{c_{\tau-1}^{(\tau-2)}}{b_{\tau,k+1}}$ and $c_i^{(\tau-1)} = c_i^{(\tau-2)} - c_{\tau-1}^{(\tau-2)} \dfrac{b_{\tau+i-(\tau-1),k+\tau-(\tau-1)}}{b_{\tau,k+\tau-(\tau-1)}}$ $\forall i = \tau, \tau + 1, \ldots, \tau + m - 1$.

Summarizing the above, where it has been shown that by sequentially substituting the delayed inputs (A.3), the coefficients $c_i^{(j)}$ are obtained as

$$\begin{gathered} c_i^{(0)} = b_{\tau+i,k+\tau} - b_{\tau,k+\tau} g_{i,k} \\ \forall i = 1, 2, \ldots, m + \tau - 1 \qquad \text{where} \qquad b_{\tau+m+j} = 0 \quad \forall j = 1, 2, \ldots \end{gathered} \tag{A.10}$$

and

$$c_i^{(j)} = \begin{cases} c_i^{(j-1)} - c_j^{(j-1)} \dfrac{b_{\tau+i-j,k+\tau-j}}{b_{\tau,k+\tau-j}} & \text{if } i \leq m + j \\ c_i^{(0)} & \text{else} \end{cases} \tag{A.11}$$

The parameters $\gamma_{i,k}$ are given by

$$\gamma_{i,k} = \frac{c_i^{(i-1)}}{b_{\tau,k+\tau-i}} \qquad \forall i = 1, 2, \ldots, \tau - 1 \tag{A.12}$$

Comparing (A.9) and (4.27), the parameters $\beta_{i,k}$ are obtained by

$$\beta_{i,k} = c_{\tau+i-1}^{(\tau-1)} \qquad \forall i = 1, 2, \ldots, m \tag{A.13}$$

A.2 Calculation of the parameters g

Solving the SDP-PIP pole-assignment problem (4.23), the controller parameters $g_{i,k}$ are not obtained directly, instead, the solution (4.31) provides $\gamma_{i,k}$, in which

the controller parameters $g_{i,k}$ are contained. Consequently, $g_{i,k}$ is required to be recovered from $\gamma_{i,k}$. In the following, it is shown that $g_{j,k}$, $j = 1, 2, \ldots, i$, are linear in $\gamma_{i,k}$, $i = 1, 2, \ldots, \tau - 1$, so that it is straightforward to perform this task and no additional burden is introduced. Similarly, the same applies to $\beta_{i,k}$.

When considering (A.12) and (A.13), it is sufficient to show that $c_i^{(j)}$ (A.11) are, indeed, linear functions w.r.t the parameters $g_{l,k}$ $\forall l = 1, 2, \ldots, i$, if $i > j$. The latter follows from the fact that in (A.12) the superscript is $j = i - 1$ and in (A.13) the subscript is $i = \tau - 1 + n$, $n = 1, 2, \ldots, m$, while the superscript is $j = \tau - 1$. Hence, in both cases, $i > j$.

So, as an exemplary case, consider $c_i^{(i-1)}$ from (A.12) and taking (A.11) into account, gives

$$
\begin{aligned}
c_i^{(i-1)} &= c_i^{(i-2)} - c_{i-1}^{(i-2)} \frac{b_{\tau+1,k+\tau-i+1}}{b_{\tau,k+\tau-i+1}} \\
&= c_i^{(i-3)} - c_{i-1}^{(i-3)} \frac{b_{\tau+1,k+\tau-i+1}}{b_{\tau,k+\tau-i+1}} \\
&\quad + c_{i-2}^{(i-3)} \left(\frac{b_{\tau+1,k+\tau-i+1} b_{\tau+1,k+\tau-i+2}}{b_{\tau,k+\tau-i+1} b_{\tau,k+\tau-i+2}} - \frac{b_{\tau+2,k+\tau-i+2}}{b_{\tau,k+\tau-i+2}} \right) \Bigg(\\
&= c_i^{(i-4)} - c_{i-1}^{(i-4)} \frac{b_{\tau+1,k+\tau-i+1}}{b_{\tau,k+\tau-i+1}} \\
&\quad + c_{i-2}^{(i-4)} \left(\frac{b_{\tau+1,k+\tau-i+1} b_{\tau+1,k+\tau-i+2}}{b_{\tau,k+\tau-i+1} b_{\tau,k+\tau-i+2}} - \frac{b_{\tau+2,k+\tau-i+2}}{b_{\tau,k+\tau-i+2}} \right) \Bigg(\\
&\quad - c_{i-3}^{(i-4)} \quad \frac{\prod_{l=1}^{3} b_{\tau+1,k+\tau-i+l}}{\prod_{l=1}^{3} b_{\tau,k+\tau-i+l}} - \frac{b_{\tau+2,k+\tau-i+2} b_{\tau+1,k+\tau-i+3}}{b_{\tau,k+\tau-i+2} b_{\tau,k+\tau-i+3}} \\
&\qquad\qquad - \frac{b_{\tau+2,k+\tau-i+3} b_{\tau+1,k+\tau-i+1}}{b_{\tau,k+\tau-i+3} b_{\tau,k+\tau-i+1}} \Bigg) \Bigg(\\
&= \ldots
\end{aligned}
\tag{A.14}
$$

where it can be seen that when proceeding until the superscript $^{(0)}$ is reached, $c_i^{(i-1)}$ is a linear function w.r.t $c_i^{(0)}, c_{i-1}^{(0)}, c_{i-2}^{(0)}, \ldots, c_1^{(0)}$. Moreover, since $c_l^{(0)}$ is a linear function w.r.t $g_{l,k}$ (A.10), it follows that, as a consequence, $c_i^{(i-1)}$ (and

hence $\gamma_{i,k}$) is a linear function w.r.t $g_{j,k}$, $j = 1, 2, \ldots, i$, as well.

Similarly, when considering (A.13) in conjunction with (A.11) and, subsequently, the observations drawn from (A.14), it can be said that $\beta_{i,k}$ is a linear function w.r.t the parameters $g_{j,k}$, $j = 1, 2, \ldots, \tau - 1 + i$.

Remark A.2.1. *It is essential that, regarding* (A.12) *and* (A.13)*,* $i > j$ *in* (A.11) *is given. Otherwise, since in each iteration the running indices,* i *and* j*, are decremented by unity (as demonstrated in* (A.14)*) the index* i *becomes zero or negative when* $j = 0$ *is reached. But, since the index* i *is associated with the index of the parameter* $g_{i,k}$ *in* (A.10)*,* i *must take a value of* $i = 1, 2, \ldots, n_g$*.*

Appendix B

B.1 The DC-DC boost converter

The purpose built laboratory based DC-DC boost converter, which is used for experiments, is shown on the right in Figure B.1. Additionally, the realisation of the load, as described in Section 5.1 and schematically shown in Figure 5.2, can be observed on the left in Figure B.1. Moreover, since the input voltage range of the ADC of the dSPACE MicroAutobox ranges between 0 V and 5 V, the output voltage of the converter is required to be scaled by a factor of $\frac{1}{4}$. This is realised by the circuit shown in the front in Figure B.1.

The Simulink® block diagram, which is used for acquiring the data that are subsequently used for obtaining the converter model, is shown in Figure B.2.

The block diagrams used for PIP control of the converter are shown in Figures B.3 and B.4. In Figure B.3, the load is considered to be of a first order transfer function form (6.1), while load steps are considered in Figure B.4. The linear PIP controllers, as well as the SDP-PIP controller are implemented as an 'Embedded MATLAB Function', hence only this function is required to be changed accordingly in order to switch between the respective controllers.

Finally, the Simulink® subsystem 'measure voltage' is shown in Figure B.5.

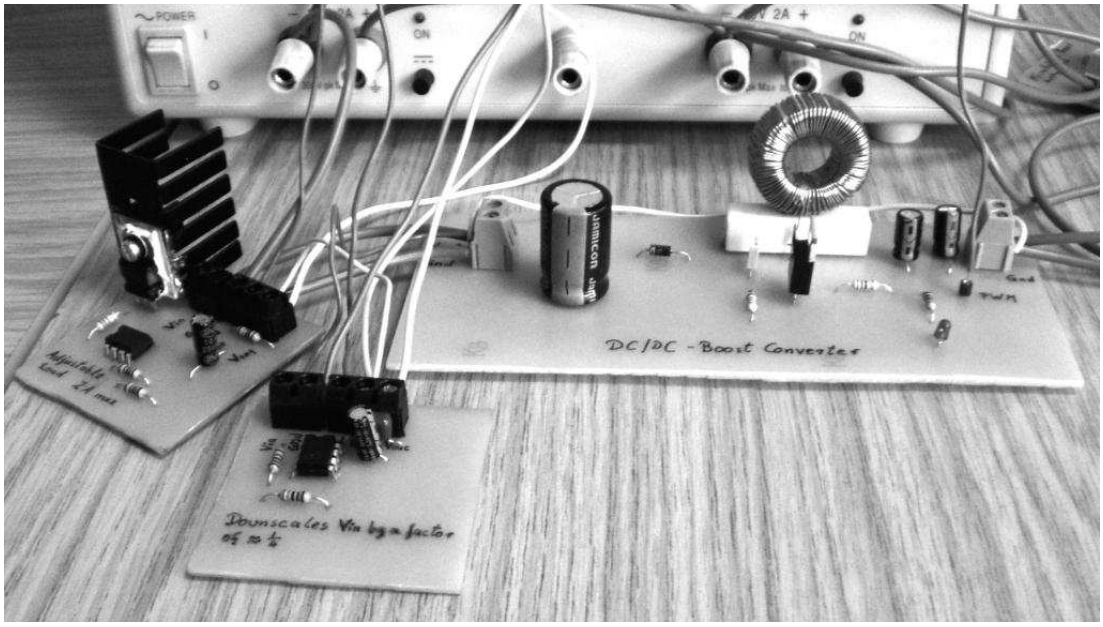
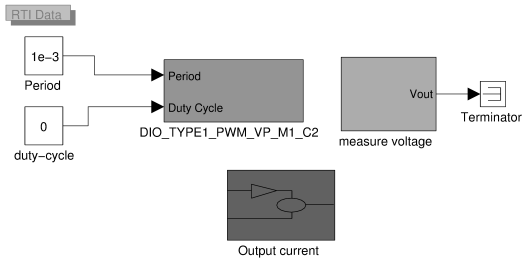
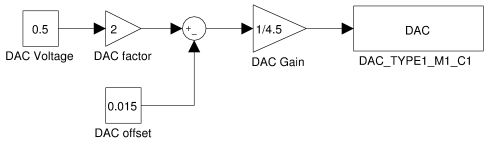


Figure B.1: DC-DC boost converter (right), realisation of the load (left) and output voltage scaling circuit (front).



(a) Simulink block diagram for acquiring open-loop data



(b) Simulink subsystem 'Output current'

Figure B.2: Simulink block diagram used for acquiring system identification data

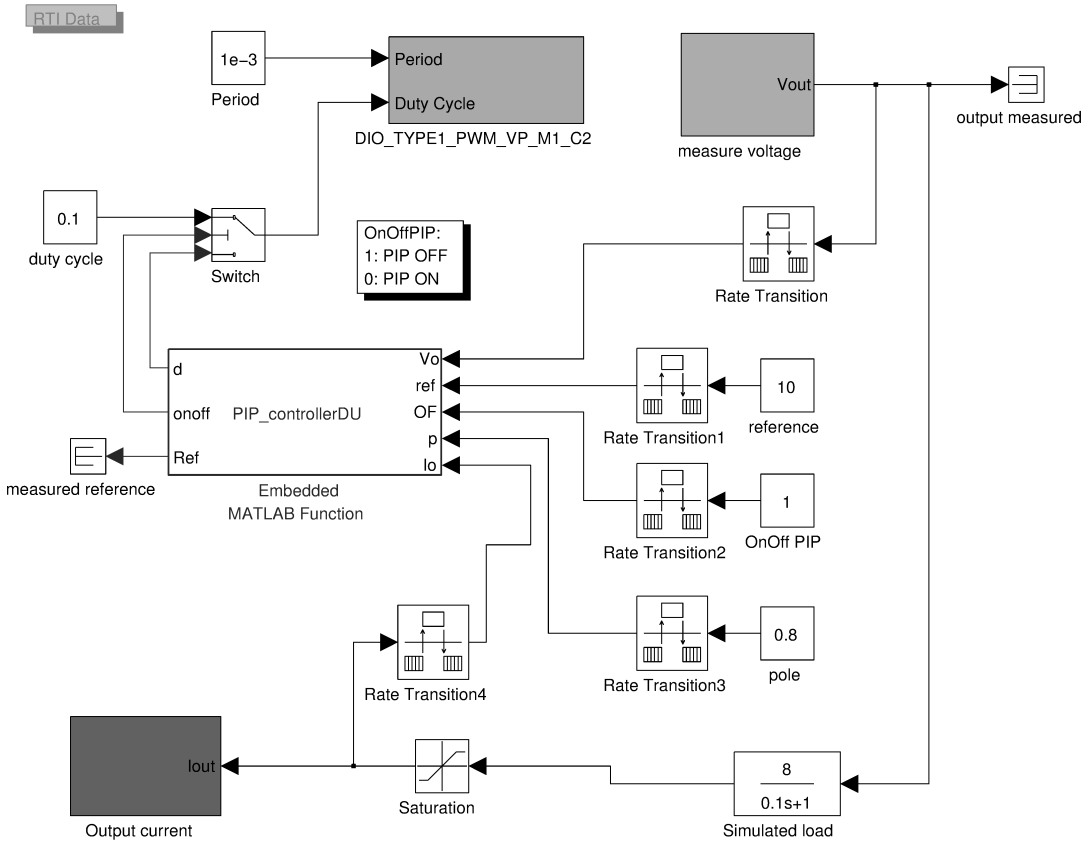


Figure B.3: Simulink block diagram when the load is realised as a transfer function.

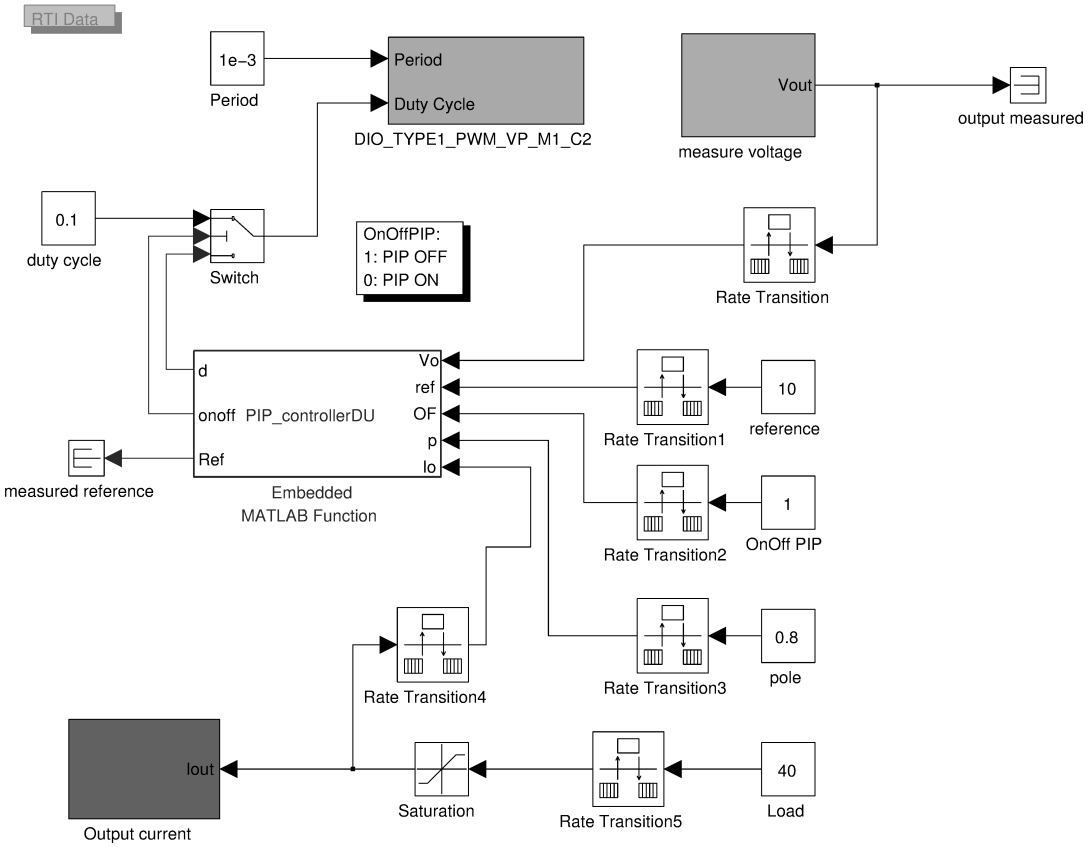


Figure B.4: Simulink block diagram when load steps are considered.

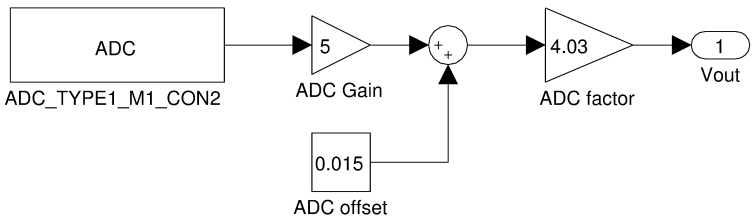


Figure B.5: Realisation of the Simulink subsystem ‘measure voltage’

References

B. M. Akesson and H. T. Toivonen. State-dependent parameter modelling and identification of stochastic non-linear sampled-data systems. *Journal of Process Control*, 16(8):877–886, 2006.

P. Albertos and A. Sala. *Multivariable Control Systems: An Engineering Approach*. Advanced Textbooks in Control and Signal Processing. Springer Verlag, London, 2004.

F. Alonge, F. D'Ippolito, F.M. Raimondi, and S. Tumminaro. Nonlinear modeling of a dc/dc converters using the hammerstein's approach. *IEEE Transactions on Power Electronics*, 22(4):1210–1221, 2007.

D. Angeli and E. Mosca. Command governors for constrained nonlinear systems. *IEEE Transactions on Automatic Control*, 44(4):816–820, 1999.

A.G Beccuti, G. Papafotiou, and M. Morari. Optimal control of the boost dc-dc converter. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 4457–4462, Seville, Spain, 2005.

A.G Beccuti, G. Papafotiou, R. Frasca, and M. Morari. Explicit hybrid model predictive control of the dc-dc boost converter. In *Proceedings of the IEEE Power Electronics Specialists Conference PESC2007*, pages 2503–2509, 2007.

A.G. Beccuti, S. Mariethoz, S. Cliquennois, S. Wang, and M. Morari. Explicit model predictive control of dc-dc switched-mode power supplies with extended kalman filtering. *IEEE Transactions on Industrial Electronics*, 56(6):1864–1874, 2009.

A. Bemporad and D. M. de la Pena. Multiobjective model predictive control. *Automatica*, 45(12):2823–2830, 2009.

A. Bemporad and M. Morari. Robust model predictive control: A survey. *Robustness in Identification and Control*, 245:207–226, 1999.

A. Bemporad and E. Mosca. Constraint fulfilment in feedback control via predictive reference management. In *Proceedings of 3rd IEEE Conference on Control Applications*, pages 1909–1914, Glasgow, 1994.

A. Bemporad, A. Casavola, and E. Mosca. Nonlinear control of constrained linear systems via predictive reference management. *IEEE Transactions on Automatic Control*, 42(3):340–349, 1997.

A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit solution of model predictive control via multiparametric quadratic programming. In *Proceedings of the American Control Conference*, pages 872–876, Chicago, USA, 2000.

A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, 2004.

S. Boyd, E. Feron, L. E. Ghaoui, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *SIAM studies in applied mathematics*. SIAM Society for Industrial and Applied Mathematics, 1994.

E. F. Camacho and C. Bordons. *Model Predictive Control*. Advanced Textbooks in Control and Signal Processing. Springer Verlag, London, 2nd edition, 2007.

M. Cannon, V. Deshmukh, and B. Kouvaritakis. Nonlinear model predictive control with polytopic invariant sets. *Automatica*, 39(8):1487–1494, 2003.

T. Cimen. State-dependent riccati equation (SDRE) control: A survey. In *Proceedings of the 17th IFAC World Congress*, pages 3761–3775, Seoul,Korea, 2008.

D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control - part I. the basic algorithm. *Automatica*, 23(2):137–148, 1987a.

D.W. Clarke, C. Mohtadi, and P.S. Tuffs. Generalized predictive control - part II. extensions and interpretations. *Automatica*, 23(2):149–160, 1987b.

J. R. Cloutier. State-dependent riccati equation techniques: An overwiew. In *Proceedings of the American Control Conference*, pages 932–936, Albuquerque, New Mexico, 1997.

D. Cortes, Jo. Alvarez, and Ja. Alvarez. Robust control of the boost converter. In *Proceedings of the International Conference on Electronics and Control Applications*, 2005.

F.B. Cunha and D.J. Pagano. Limitations in the control of a dc-dc boost converter. In *Proceedings of the 15th IFAC Triennal World Congress*, Barcelona, Spain, 2002.

A. Davoudi and J. Jatskevich. Parasitics realization in state-space average-value modelling of pwm dc-dc converters using an equal area method. *IEEE Transactions on Power Electronics*, 54(9):1960–1967, 2007.

A. Davoudi, J. Jatskevich, and T. De Rybel. Numerical state-space average-value modeling of pwm dc-dc converters operating in dcm and ccm. *IEEE Transactions on Power Electronics*, 21(4):1003–1012, 2006.

R. W. Erickson and M. Maksimovic. *Fundamentals of Power Electronics*. Kluwer Academics/Plenum Publishers, New York, 2nd edition, 2001.

V. Exadaktylos. *Model Predictive Control Structures in the Non-Minimal State-Space*. PhD thesis, Lancaster University, 2007.

V. Exadaktylos and C. J. Taylor. Multi-objective performance optimisation for model predictive control by goal attainment. *International Journal of Control*, 83(7):1374–1386, 2010.

V. Exadaktylos, C. J. Taylor, and A. Chotai. Model predictive control using a non-minimal state space form with an integral-of-errors state variable. In *Proceedings of UKACC International Conference on Control 06*, Paper-72, Glasgow, UK, 2006.

V. Exadaktylos, C. J. Taylor, and A. Chotai. Constraint handling for state dependent parameter models. In *Proceedings of UKACC International Conference on Control 08*, Manchester, UK, 2008.

H. El Fadil and F. Giri. Robust and nonlinear control of pwm dc-to-dc boost power converters. In *Proceedings of the IEEE Power Electronics Specialists Conference, PESC2007*, pages 407–412, 2007.

R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons Ltd., 2nd edition, 2000.

P. Gahinet and A. Nemirovski. The projective method for solving linear matrix inequalities. *Mathematical Programming*, 77(1):163–190, 1997.

Z. Gajic and M. Lelic. *Modern Control Systems Engineering*. Prentice Hall, Hertfordshire, 1996.

T. Geyer, G. Papafotiou, R. Frasca, and M. Morari. Constrained optimal control of the step-down dc-dc converter. *IEEE Transactions on Power Electronics*, 23 (5):2454–2464, 2008.

E. G. Gilbert and I. Kolmanovsky. Discrete-time reference governors and the nonlinear control of systems with state and control constraints. *International Journal of Robust and Nonlinear Control*, 5:487–504, 1995.

G. C. Goodwin, M. M. Seron, and J. A. De Dona. *Constrained Control and Estimation: An Optimisation Approach*. Communications and Control Engineering. Springer Verlag, London, 2005.

D. J. Griffiths. *Introduction to Electrodynamics*. Prentice Hall, New Jersey, 3rd edition, 1999.

S. Gunnarsson, V. Collignon, and O. Rousseaux. Tuning of a decoupling controller for a 2x2 system using iterative feedback tuning. *Control Engineering Practice*, 11(9):1035–1041, 2003.

C.J. Harris and S.A. Billings. *Self-Tuning and Adaptive Control: Theory and Applications*. IEE Control Engineering Series Vol. 15. Peter Perengrinus Ltd., Stevenage, UK, 1981.

T. Hesketh. State-space pole-placing self-tuning regulator using input-output values. *IEE Proceedings D Control Theory and Applications*, 129(4):123–128, 1982.

F. S. Hillier and G. J. Lieberman. *Introduction to Operations Research*. McGraw-Hill, New York, 7th edition, 2001.

U. Hitzemann. Output voltage control of a dc-dc switch converter. Msc thesis, Coventry University, UK, 2009.

U. Hitzemann and K. J. Burnham. State dependent parameter modelling and control of a dc-dc boost converter in discontinuous conduction mode. In *Proceedings of the 9th European Workshop on Advanced Control and Diagnosis, ACD 2011*, Paper-56, Budapest, Hungary, 2011.

U. Hitzemann and K.J. Burnham. Decoupling model predictive control in a non-minimal state space representation. In *Proceedings of the 8th European Workshop on Advanced Control and Diagnosis, ACD 2010*, Ferrara,Italy, 2010.

U. Hitzemann and K.J. Burnham. State dependent parameter modelling of a dc-dc boost converter operating in discontinuous conduction mode. In *Proceedings of the 9th International Conference on Informatics in Control, Automation and Robotics, ICINCO 2012*, pages 482–487, Rome,Italy, 2012.

P. Horowitz and W. Hill. *The Art of Electronics*. Cambridge University Press, New York, 2nd edition, 1989.

T. C. Hsia. *System Identification: Least-Squares Methods*. Lexington Books, Massachusetts, 1977.

E. Ikonen and K. Najim. *Advanced Process Identification and Control*. Marcel Dekker Inc., New York, 2002.

C. M. Ivan, D. Lascu, and V. Popescu. A new averaged switch model including conduction losses for pwm converters operating in discontinuous inductor current mode. *Facta Universitatis Ser.: Elec. Energ.*, 19(2):219–230, 2006.

K. H. Johansson. The quadruple tank process: A multivariable laboratory process with an adjustable zero. *IEEE Transactions on Control Systems Technology*, 8 (3):456–465, 2000.

K. H. Johansson. Interaction bounds in multivariable control systems. *Automatica*, 38(6):1045–1051, 2002.

K. H. Johansson and J. L. R. Nunes. A multivariable laboratory process with an adjustable zero. In *Proceedings of the 17th American Control Conference*, Philadelphia, PA, 1998.

T. Kailath. *Linear Systems*. Prentice Hall, New Jersey, 1980.

N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

P. Kontoroupis, P. C. Young, A. Chotai, and C. J. Taylor. State dependent parameter-proportional integral plus (sdp-pip) control of nonlinear systems. In *Proceedings of 16th International Conference on Systems Engineering, ICSE'2003*, pages 373–378, Coventry,UK, 2003.

M. V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequalities. *Automatica*, 32(10):1361–1379, 1996.

B. Kouvaritakis, J. A. Rossiter, and J. Schuurmans. Efficient robust predictive control. *IEEE Transaction on Automatic Control*, 45(10):1545–1549, 2000.

B. Kouvaritakis, M. Cannon, and J. A. Rossiter. Who needs QP for linear MPC anyway? *Automatica*, 38(5):879–884, 2002.

M. Kubalcik and V. Bobal. Adaptive control of coupled-drives apparatus based on polynomial theory. *Proc. IMechE Part I: Journal of Systems and Control Engineering*, 220(7):641–654, 2006.

W. H. Kwon and S. Han. *Receding Horizon Control: Model Predictive Control for State Models*. Advanced Textbooks in Control and Signal Processing. Springer Verlag, Leipzig, 2005.

T. Larkowski and K.J. Burnham. Bilinear approach to modelling of continuous stirred tank reactor process. In *Proceedings of the 9th European Workshop on Advanced Control and Diagnosis, ACD 2011*, Paper-54, Budapest, Hungary, 2011.

T. Larkowski, U. Hitzemann, and K.J. Burnham. Modelling and identification of a dc-dc boost converter operating in discontinuous conduction mode. In *Proceedings of the IET Control and Automation Conference*, Birmingham, UK, 2013.

M. J. Lees, P. C. Young, A. Chotai, and W. Tych. A non-minimal state variable feedback approach to multivariable control of glasshouse climate. *Trans. Inst. Measurement and Control*, 17(4):200–211, 1995.

C. Lin and T. Hsieh. Decoupling controller design for linear multivariable plants. *IEEE Transactions on Automatic Control*, 36(4):485–489, 1991.

Lennart Ljung. *System Identification – Theory for the User*. Prentice Hall PTR, New York, 2nd edition, 1999.

J.M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2001.

D. Maksimovic and S. Cuk. A unified analysis of pwm converters in discontinuous modes. *IEEE Transactions on Power Electronics*, 6(3):476–490, 1991.

D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

A. McCabe, P. C. Young, A. Chotai, and C. J. Taylor. Proportional-integral-plus (pip) control of non-linear systems. In *Proceedings of the International Conference on Systems Engineering ICSE2000*, pages 407–412, Coventry, UK, 2000.

R. D. Middlebrook and S. Cuk. A general unified approach to modelling switching-converter power stages. In *IEEE Proceedings of Power Electronics Specialists Conference*, pages 18–34, 1976.

N. Mohan, T.M. Undeland, and W.P. Robbins. *Power Electronics - Converters, Applications, and Design*. John Wiley and Sons Inc., USA, 2nd edition, 1995.

M. Morari and J.H. Lee. Model predictive control: Past, present and future. *Computers & Chemical Engineering*, 23(4):667–682, 1999.

O. Nelles. *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*. Springer Verlag, Berlin, 2001.

A. Nemirovskii and P. Gahinet. The projective method for solving linear matrix inequalities. In *Proceedings of the American Control Conference*, pages 840–844, Baltimore, Maryland, 1994.

Y. Nesterov and A. Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM Studies in Applied and Numerical Mathematics; Vol. 13. SIAM Society for Industrial and Applied Mathematics, Philadelphia, 1994.

RWD Nickalls. Vieta, descarte and the cubic equation. *The Mathematical Gazette*, 90(518):203–208, 2006.

Norman S. Nise. *Control Systems Engineering*. John Wiley and Sons (Asia) Pte Ltd, Asia, 5th edition, 2008.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Verlag, New York, 2nd edition, 2006.

Katsuhiko Ogata. *Modern Control Engineering*. Prentice Hall, New Jersey, 3rd edition, 1997.

C. Olalla, R. Leyva, A. El Aroudi, and I. Queinnec. Robust lqr control for pwm converters: An lmi approach. *IEEE Transactions on Industrial Electronics*, 56 (7):2548–2558, 2009.

C. Olalla, R. Leyva, A. El Aroudi, P. Garces, and I. Queinnec. Lmi robust control design for boost pwm converters. *IET Power Electronics*, 3(1):75–85, 2010.

C. Olalla, I. Queinnec, R. Leyva, and A. El Aroudi. Robust optimal control of bilinear dc-dc converters. *Control Engineering Practice*, 19(7):688–699, 2011.

A. V. Oppenheim, R. W. Schafer, and J. R. Buck. *Discrete-time signal Processing*. Prentice Hall, New Jersey, 2nd edition, 1998.

A. R. Plummer and N. D. Vaughan. Decoupling pole-placement control, with application to a multi-channel electro-hydraulic servosystem. *Control Engineering Practice*, 5(3):313–323, 1997.

C. Qiao and J. Zhang. Control of boost type converter in discontinuous conduction mode by controlling the product of inductor voltage-second. In *Proceedings of the 36th Power Electronics Specialists Conference, PESC'05*, pages 1213–1219, Recife, Brazil, 2005.

S. J. Qin and T. A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7):733–764, 2003.

J. Renegar. *A Mathematical view of interior-point methods in convex optimization*. SIAM Society for Industrial and Applied Mathematics, Philadelphia, 2001.

J. A. Rossiter. *Model-based Predictive Control: A Practical Approach*. CRC Press, Boca Raton, 2004.

S. R. Sanders, J.M. Noworolski, X.Z. Liu, and G.C. Verghese. Generalized averaging method for power conversion circuits. *IEEE Transactions on Power Electronics*, 6(2):251–259, 1991.

E. M. Shaban and C. J. Taylor. Proportional-integral-plus control of a class of nonlinear systems using exact and partial linearisaton by feedback. *ICGST International Journal on Automatic Control and Systems Engineering, ACSE*, 6:55–70, 2006.

H. Sira-Ramirez, M.A. Oliver-Salazart, J. A. Vazquez-Santacruz, and M. Velasco-Villar. On the robust control of the boost converter. In *Proceedings of the 37th Annual Conference on IEEE Industrial Electronics Society, IECON2011*, pages 1497–1502, 2011.

O. J. M. Smith. A controller to overcome dead time. *ISA Journal*, 6(2):28–33, 1959.

D. D. Sourlas. Optimization-based decoupling controller design for discrete systems. *Chemical Engineering Science*, 56(15):4695–4710, 2001.

M. A. Stables and C. J. Taylor. Non-linear control of ventilation rate using state-dependent parameter models. *Biosystems Engineering*, 95(1):7–18, 2006.

M. A. Stables, C. J. Taylor, and A. Chotai. Control of micro-climate using time delay state dependent parameter models. In *Proceedings of the 18th International Conference on Systems Engineering (ICSE)*, 2006.

J. Sun, D. M. Mitchell, M. F. Greuel, P. T. Krein, and R. M. Bass. Averaged modeling of pwm converters operating in discontinuous conduction mode. *IEEE Transactions on Power Electronics*, 16(4):482–492, 2001.

T. Söderström and P. Stoica. *System Identification*. Systems and Control Engineering. Prentice Hall International, Cambridge, UK, 1989.

C. J. Taylor, P. C. Young, and A. Chotai. PIP optimal control with a risk sensitive criterion. In *Proceedings of UKACC International Conference on Control 96*, 1996a.

C. J. Taylor, P. C. Young, A. Chotai, W. Tych, and M. J. Lee. The importance of structure in PIP control design. In *Proceedings of UKACC International Conference on Control 96*, 1996b.

C. J. Taylor, A. Chotai, and P. C. Young. State space control system design based on non-minimal state-variable feedback: further generalization and unification results. *International Journal of Control*, 73(14):1329–1345, 2000.

C. J. Taylor, A. Chotai, and P.C. Young. Design and application of pip controllers: robust control of the IFAC93 benchmark. *Transactions of the Institute of Measurement and Control*, 23(3):183–200, 2001.

C. J. Taylor, D. J. Pedegral, P. C. Young, and W. Tych. Environmental time series analysis and forecasting with the Captain toolbox. *Environmental Modelling and Software*, 22(6):797–814, 2007a.

C. J. Taylor, E. M. Shaban, M. A. Stables, and S. Ako. Proportional-integral-plus control applications of state-dependent parameter models. *Proc. IMechE Part I: Journal of Systems and Control Engineering*, 221(7):1019–1032, 2007b.

C. J. Taylor, A. Chotai, and P. C. Young. Non-linear control by input-output state variable feedback pole assignment. *International Journal of Control*, 82 (6):1029–1044, 2009.

C.J. Taylor, A. Chotai, and K.J. Burnham. Controllable forms for stabilising pole assignment design of generalised bilinear systems. *Electronics Letters*, 47 (7):431–439, 2011.

C.J. Taylor, A. Chotai, and P. Cross. Non-minimal state variable feedback decoupling control for multivariable continuous-time systems. *International Journal of Control*, 85(6):722–734, 2012.

C.K. Tse and K.M. Adams. Qualitative analysis and control of a dc-to-dc boost converter operating in discontinuous mode. *IEEE Transactions on Power Electronics*, 5(3):323–330, 1990.

C. L. Wang and P. C. Young. Direct digital and adaptive control by input-output state variable feedback pole assignment. *International Journal of Control*, 47 (1):97–109, 1988.

L. Wang and P. C. Young. An improved structure for model predictive control using non-minimal state space realisation. *Journal of Process Control*, 16(4): 355–371, 2006.

Liuping Wang. *Model Predictive Control System Design and Implementation using MATLAB*. Springer Verlag, London, 2009.

M. Wei, Q. Wang, and X. Cheng. Some new results for system decoupling and pole assignment problems. *Automatica*, 46(5):937–944, 2010.

P.E. Wellstead and M.B. Zarrop. *Self-Tuning Systems: Control and Signal Processing*. John Wiley & Sons Ltd., West Sussex, England, 1991.

M. Wens and M. Steyaert. *Design and Implementation of Fully-Integrated Inductive DC-DC Converters in Standard CMOS*. Analog Circuits and Signal Processing. Springer Verlag, 2011.

S. J. Wright. *Primal-Dual Interior-Point Methods*. SIAM Society for Industrial and Applied Mathematics, Philadelphia, 1997.

G. Xie, H. Fang, and X. Cheng. Non-ideal models and simulation of boost converters operating in dcm. *International Journal of Computer and Electrical Engineering*, 2(4):730–733, 2010.

P. C. Young. Stochastic, dynamic modelling and signal processing: time variable and state dependent parameter estimation. In W. C. Fitzgerald, R. L. Smith, A. T. Walden, and P. C. Young, editors, *Nonlinear and Nonstationary Signal Processing*, pages 74–114. Cambridge University Press, Cambridge, 2000.

P. C. Young. *Recursive Estimation and Time Series Analysis - An Introduction for the Student and Practitioner*. Springer Verlag, Berlin, 2nd edition, 2011.

P. C. Young, M. A. Behzadi, C. L. Wang, and A. Chotai. Direct digital and adaptive control by input-output state variable feedback pole assignment. *International Journal of Control*, 46(6):1867–1881, 1987.

P.C. Young and C. J. Taylor. Recent developments in the CAPTAIN toolbox for Matlab. In *Proceedings of IFAC SYSID Conference*, pages 1838–1843, Brussels, Belgium, 2012.

P.C. Young, P. Mckenna, and J. Bruun. Identification of non-linear stochastic systems by state dependent parameter estimation. *International Journal of Control*, 74(18):1837–1857, 2001.

S. J. Ziemian. *Bilinear Proportional-Integral-Plus Control*. PhD thesis, Coventry University, 2002.